

PENENTUAN PENERIMAAN BEASISWA MENGGUNAKAN METODE MODIFIED K-NEAREST NEIGHBOR

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Caesaredi Rama Raharya

NIM: 145150200111141



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

PENETUAN PENERIMAAN BEASISWA MENGGUNAKAN METODE *MODIFIED K-NEAREST NEIGHBOR*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:

Caesaredi Rama Raharya

NIM: 145150200111141

Skripsi ini telah diuji dan dinyatakan lulus pada

6 Juni 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Nurul Hidayat, S.Pd, M.Sc

NIP. 19680430 200212 1 001

Edy Santoso, S.Si, M.Kom

NIP. 19740414 200312 1 004

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP. 19710518 200312 1 001

PERNYATAAN ORISINALITAS

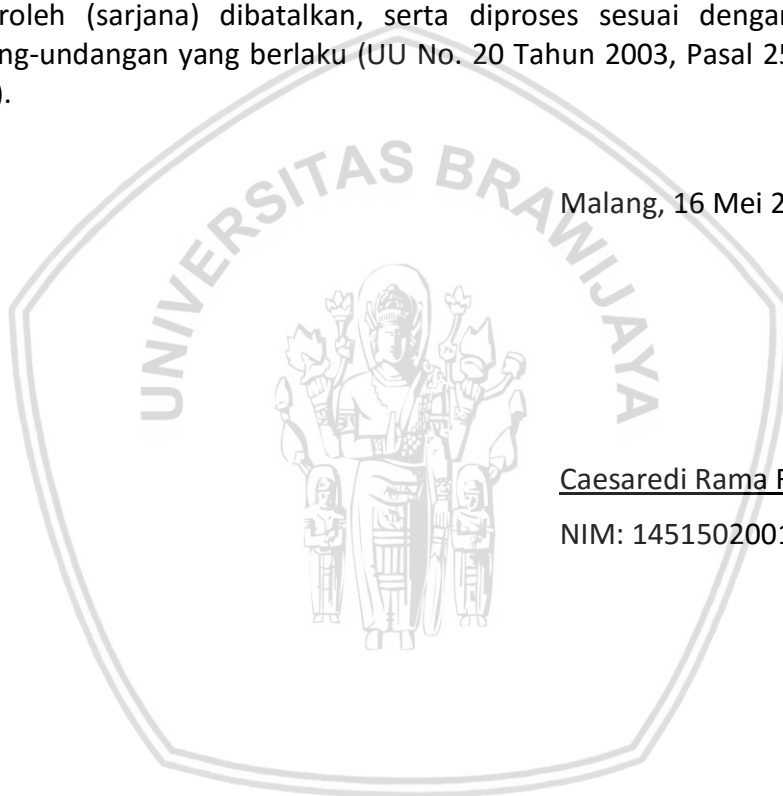
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 16 Mei 2018

Caesaredi Rama Raharya

NIM: 145150200111141



KATA PENGANTAR

Puja dan puji syukur kepada Tuhan Yang Maha Esa karena berkat rahmat, taufik serta hidayah-Nya penulis dapat menyelesaikan laporan skripsi yang berjudul “Penentuan Penerimaan Beasiswa Menggunakan Metode Modified K-Nearest Neighbor”. Penulis menyadari bahwa skripsi ini tidak akan berhasil tanpa bantuan dari pihak-pihak yang memberikan bantuan terhadap penyelesaian laporan skripsi ini, oleh karena itu penulis ingin mengucapkan terima kasih kepada:

1. Nurul Hidayat, S.Pd., M.Sc., dan Edy Santoso, S.Si., M.Kom., selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan laporan skripsi ini.
2. Tri Astoto Kurniawan, S.T., M.T., Ph.D., selaku Ketua Jurusan Program Studi Teknik Informatika.
3. Agus Wahyu Widodo, S.T., M.Cs., selaku Ketua Program Studi Teknik Informatika.
4. Kedua orangtua penulis yang selalu memberikan semangat motivasi, serta senantiasa memberikan doa demi terselesaikannya skripsi ini.
5. Seluruh teman-teman penulis Afandi, Kevin, Ridho, Bagus, Gilang, Yusuf, Aldo, Hafid dan lain-lain. Yang selalu memberikan semangat untuk segera menyelesaikan laporan skripsi ini.
6. Kedua saudara kandung penulis edita rosana dan edinar valiant yang memberikan arahan dan semangat.

Penulis sangat menyadari bahwa laporan skripsi yang penulis susun ini masih jauh dari kata sempurna. Penulis mohon maaf apabila ada kesalahan dan kekurangan dalam laporan skripsi ini. Demikian laporan skripsi yang penulis susun, akhir kata penulis ucapkan terima kasih yang sebanyak-banyaknya dan berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 16 Mei 2018

Penulis
Caesaredi Rama Raharya
caesarediraharya@gmail.com

ABSTRAK

Dalam menentukan penerimaan beasiswa terhadap pengajuan beasiswa mahasiswa, petugas seleksi beasiswa seringkali mengalami permasalahan kesulitan penentuan mahasiswa yang mendapat beasiswa, yang disebabkan oleh beberapa faktor seperti banyaknya jumlah mahasiswa yang mendaftar untuk mendapatkan beasiswa sedangkan kuota untuk mahasiswa yang mendapatkan beasiswa relatif sedikit, banyaknya parameter yang digunakan sebagai acuan dalam penentuan mahasiswa yang berhak mendapat beasiswa dan petugas seleksi beasiswa yang hanya diberi waktu yang relatif singkat dalam penentuan mahasiswa yang mendapat beasiswa. Sehingga diperlukan sebuah sistem klasifikasi yang dapat diimplementasikan dalam persoalan tersebut untuk memudahkan petugas seleksi beasiswa. Penelitian ini dilakukan dengan dataset sebanyak 370 data pengajuan beasiswa dengan 5 parameter dan 2 status penerimaan menggunakan metode *Modified K-Nearest Neighbor*. Metode *Modified K-Nearest Neighbor* terdiri dari proses normalisasi, jarak *euclidean*, validitas dan pembobotan *weight voting*. Keluaran dari sistem ini adalah berupa hasil penerimaan beasiswa. Hasil rata-rata akurasi tertinggi yang didapatkan berdasarkan pengujian yang telah dilakukan adalah sebesar 87.2%. Dari hasil akurasi tersebut, maka dapat disimpulkan metode *Modified K-Nearest Neighbor* dapat diimplementasikan ke dalam sistem penentuan penerimaan beasiswa.

Kata kunci: *Modified K-Nearest Neighbor*, Klasifikasi, Penerimaan Beasiswa

ABSTRACT

In determining the acceptance of students scholarship, the officers often face problem selecting student who are eligible for a scholarship that caused by several factors such as the number of students who apply for a scholarship while the quota for students who get scholarships is relatively small, the number of parameters used as a reference in determining the students who are eligible for a scholarship and the officers who are given only a relatively short time in determining the awardee. Therefore implement a classification system is required to this issue for help facilitate the scholarship selection officer. This research was conducted with dataset as much as 370 data of scholarship submission with 5 parameters and 2 acceptance status using Modified K-Nearest Neighbor method. Modified Method K-Nearest Neighbor consists of the process of calculating the normalization, distance euclidean, calculation of validity value and weight loss calculation. the output of this system is the result of the scholarship receipt. The highest average accuracy results obtained based on the tests that have been done is 87.2%. From the result of accuracy, it can be concluded that the method of Modified K-Nearest Neighbor method can be implemented into the system of determination schorlaship acceptance.

Keywords : *Modified K-Nearest Neighbor, Classification, Scholarship grante*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
DAFTAR PERSAMAAN.....	xii
DAFTAR SOURCE CODE	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian pustaka	5
2.2 Data Mining.....	6
2.3 Normalisasi data	6
2.4 Algoritme K-Nearest Neighbor	7
2.5 Algoritme Modified K-Nearest Neighbor.....	7
2.5.1 Validitas data.....	8
2.5.2 Weight voting.....	8
2.6 Akurasi sistem	9
BAB 3 METODOLOGI	10
3.1 Studi literatur	10
3.2 Analisis kebutuhan sistem	11
3.2.1 Kebutuhan Fungsional:	11

3.2.2 Kebutuhan Non-Fungsional:	11
3.3 Pengumpulan data	11
3.4 Pengolahan data	12
3.5 Perancangan sistem	12
3.6 Implementasi sistem	12
3.7 Pengujian	12
3.8 Analisis	12
3.9 Kesimpulan	13
BAB 4 Perancangan sistem	14
4.1 Deskripsi umum sistem	14
4.2 Deskripsi data	15
4.3 Perancangan sistem	15
4.3.1 Perancangan proses	16
4.3.2 Proses normalisasi data	16
4.3.3 Proses klasifikasi MKNN	17
4.3.4 Perhitungan jarak Euclidean	18
4.3.5 Perhitungan validitas	19
4.3.6 Perhitungan weight voting	21
4.4 Manualisasi	21
4.4.1 Menentukan nilai k	23
4.4.2 Normalisasi data	23
4.4.3 Perhitungan jarak Euclidean data latih	24
4.4.4 Validitas data	25
4.4.5 Jarak Euclidean data latih dan data uji	26
4.4.6 Weight voting	27
4.5 Perancangan antarmuka	28
4.5.1 Antarmuka halaman utama	28
4.5.2 Antarmuka halaman hasil penerimaan beasiswa	29
4.6 Perancangan Pengujian	30
4.6.1 Rancangan pengujian pengaruh nilai k	30
4.6.2 Rancangan pengujian kelas seimbang dan tidak seimbang	30
4.6.3 Rancangan pengujian pengaruh jumlah data latih	31

BAB 5 implementasi	32
5.1 Implementasi sistem.....	32
5.1.1 Spesifikasi perangkat keras	32
5.1.2 Spesifikasi perangkat lunak.....	33
5.2 Batasan implementasi	33
5.3 Implementasi algoritme.....	33
5.3.1 Implementasi perhitungan normalisasi data	34
5.3.2 Implementasi algoritme perhitungan Euclidean.....	37
5.3.3 Implementasi algoritme perhitungan validitas.....	40
5.3.4 Implementasi algoritme perhitungan weight voting	41
5.4 Implementasi antarmuka.....	44
5.4.1 Implementasi antarmuka halaman utama.....	44
5.4.2 Implementasi antarmuka halaman hasil penerimaan beasiswa	45
BAB 6 Pengujian dan analisis	46
6.1 Pengujian pengaruh nilai k.....	46
6.1.1 Skenario pengujian pengaruh nilai k.....	46
6.1.2 Analisis hasil pengujian pengaruh nilai k	46
6.2 Pengujian kelas seimbang dan tidak seimbang	50
6.2.1 Skenario pengujian kelas seimbang dan tidak seimbang	50
6.2.2 Analisis hasil pengujian kelas seimbang dan tidak seimbang	50
6.3 Pengujian pengaruh jumlah data latih.....	52
6.3.1 Skenario pengujian pengaruh jumlah data latih.....	52
6.3.2 Analisis hasil pengujian pengaruh jumlah data latih	53
BAB 7 Penutup	55
7.1 Kesimpulan.....	55
7.2 Saran	56
DAFTAR PUSTAKA.....	57
LAMPIRAN	59

DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	5
Tabel 4.1 Parameter dan kode parameter.....	15
Tabel 4.2 Dataset	22
Tabel 4.3 Data Latih	22
Tabel 4.4 Data Uji	23
Tabel 4.5 Normalisasi Data Latih.....	24
Tabel 4.6 Normalisasi Data Uji	24
Tabel 4.7 Jarak <i>Euclidean</i> data latih	25
Tabel 4.8 Validitas Data.....	25
Tabel 4.9 Jarak <i>Euclidean</i> data uji dan data latih.....	26
Tabel 4.10 <i>Weight Voting</i>	27
Tabel 4.11 Tabel penentuan kelas hasil	28
Tabel 4.12 Rancangan Pengujian Pengaruh Nilai K.....	30
Tabel 4.13 Rancangan Pengujian Kelas Seimbang Dan Tidak Seimbang	30
Tabel 4.14 Rancangan Pengujian Pengaruh Jumlah Data Latih	31
Tabel 5.1 Spesifikasi Perangkat Keras	33
Tabel 5.2 Spesifikasi Perangkat Lunak	33
Tabel 6.1 Hasil pengujian pengaruh nilai k	47
Tabel 6.2 Rata-rata hasil pengujian pengaruh nilai k.....	48
Tabel 6.3 Hasil pengujian kelas seimbang dan tidak seimbang	50
Tabel 6.4 Rata-rata hasil pengujian kelas seimbang dan tidak seimbang	51
Tabel 6.5 Hasil pengujian pengaruh jumlah data latih	53
Tabel 6.6 Rata-rata hasil pengujian pengaruh jumlah data latih.....	53

DAFTAR GAMBAR

Gambar 3.1 Diagram alir metodologi.....	10
Gambar 4.1 Diagram alir perancangan sistem.....	14
Gambar 4.2 Diagram alir proses sistem	16
Gambar 4.3 Diagram alir normalisasi data	17
Gambar 4.4 Diagram alir proses klasifikasi MKNN.....	18
Gambar 4.5 Diagram alir Perhitungan Jarak <i>Euclidean</i>	19
Gambar 4.6 Diagram alir Perhitungan <i>Validitas</i>	20
Gambar 4.7 Diagram alir Perhitungan <i>Weight Voting</i>	21
Gambar 4.8 Perancangan antarmuka halaman utama.....	29
Gambar 4.9 Perancangan antarmuka halaman hasil.....	29
Gambar 5.1 Diagram alir implementasi	32
Gambar 5.2 Implementasi Antarmuka Halaman Utama	45
Gambar 5.3 Implementasi antarmuka halaman hasil penerimaan beasiswa	45
Gambar 6.1 Diagram alir pengujian dan analisis	46
Gambar 6.2 Grafik hasil pengujian pengaruh nilai k.....	49
Gambar 6.3 Grafik hasil pengujian kelas seimbang dan tidak seimbang	52
Gambar 6.4 Grafik hasil pengujian pengaruh jumlah data latih	54

DAFTAR PERSAMAAN

Persamaan 2.1 Min-max Normalization	7
Persamaan 2.2 Jarak Euclidean	7
Persamaan 2.3 Validitas	8
Persamaan 2.4 Weight Voting	8
Persamaan 2.5 Akurasi Sistem	9



DAFTAR SOURCE CODE

Source code 5.1 Normalisasi data.....	36
Source code 5.2 Jarak euclidean	39
Source code 5.3 Validitas	41
Source code 5.4 Weight voting	43



BAB 1 PENDAHULUAN

1.1 Latar belakang

Berdasarkan data dari Badan Pusat Statistik pada tahun 2018 total ada 48,81% wanita dan 38,81% laki-laki berusia 19-24 tahun di Kota Malang tidak melanjutkan sekolah ke perguruan tinggi dikarenakan banyak faktor, salah satu faktor utamanya adalah tingginya biaya perguruan tinggi (Mazda, 2018). Dari persoalan tersebut sebenarnya ada solusi berupa beasiswa. Beasiswa adalah merupakan pemberian bantuan keuangan maupun pendidikan yang diberikan kepada mahasiswa atau pelajar. Yang digunakan demi keberlangsungan pendidikan yang ditempuh dengan syarat telah memenuhi syarat-syarat yang ditentukan dari pihak pemberi beasiswa (Sulistyo, 2015). Berdasarkan hasil survei yang dilakukan oleh penulis pada salah satu perguruan tinggi swasta di Kota Malang yang memberikan beasiswa kepada mahasiswanya yaitu Universitas Wisnuwardhana Malang. Dalam menentukan pemberian beasiswa kepada mahasiswa, petugas seleksi beasiswa seringkali mengalami permasalahan kesulitan menentukan mahasiswa yang mendapat beasiswa, yang disebabkan oleh beberapa faktor seperti banyaknya jumlah mahasiswa yang mendaftar untuk mendapatkan beasiswa sedangkan kuota untuk mahasiswa yang mendapatkan beasiswa relatif sedikit, banyaknya parameter yang digunakan sebagai acuan dalam penentuan mahasiswa yang berhak mendapat beasiswa dan petugas seleksi beasiswa yang hanya diberi waktu yang relatif singkat dalam penentuan mahasiswa yang mendapat beasiswa. Sehingga diperlukan sebuah sistem klasifikasi yang dapat diimplementasikan dalam persoalan tersebut untuk memudahkan petugas seleksi beasiswa.

Dalam penelitian mengenai klasifikasi penerimaan beasiswa ini dikarenakan data yang didapatkan dari Universitas Wisnuwardhana Malang tidak valid dikarenakan pihak petugas seleksi beasiswa hanya menyimpan data mahasiswa yang mendapat beasiswa sedangkan pengajuan beasiswa yang ditolak data tersebut tidak disimpan sehingga tidak dapat digunakan untuk penelitian. Data yang digunakan didapatkan dari penelitian skripsi sebelumnya yang telah valid yaitu data dari penerimaan beasiswa Toyota Astra tahun 2015 dan 2016, Yayasan Toyota Astra (YTA) memberikan kesempatan untuk mahasiswa dapat mengajukan diri sebagai calon penerima beasiswa. Penentuan diterima maupun ditolak bisa diketahui berdasarkan syarat-syarat utama yang diajukan pihak YTA yaitu nilai IPK minimal 3,00, mahasiswa berkebangsaan Indonesia, khusus menekuni bidang studi Teknik, tidak sedang menerima beasiswa lain dan mahasiswa berada di semester V/VII. Dimana, data telah memiliki parameter yang sesuai dengan penelitian. Penelitian ini mendapat referensi dan data dari penelitian yang pernah dilakukan oleh (Lestari, 2017) dimana dari penelitian tersebut diketahui bahwa dalam metode klasifikasi dalam penerimaan beasiswa antara metode *Naïve Bayes* dan metode *K-Nearest Neighbor* (KNN), tingkat akurasi dari metode KNN lebih baik. Selanjutnya pada penelitian yang dilakukan oleh (Parvin, et al., 2008) untuk

mencari perbandingan antara metode KNN dengan metode *Modified K-Nearest Neighbor*(MKNN) didapatkan hasil bahwa nilai akurasi dari metode MKNN lebih baik dari metode KNN. Dengan asumsi tersebut penelitian ini menggunakan metode MKNN dalam kasus permasalahan penentuan penerimaan beasiswa.

Penelitian lain sebelumnya menggunakan metode MKNN pada penelitian dari (Wafiyah, 2017) yaitu yang berjudul “Implementasi Algoritma *Modified K-Nearest Neighbor* (MKNN) Untuk Klasifikasi Penyakit Demam”, pengujian dilakukan dengan 3 tahap yaitu yang pertama terhadap nilai K mendapat rata-rata akurasi sebesar 88.55% , lalu pengujian pada jumlah data latih dengan rata-rata sebesar 92.42%, yang terakhir pengujian pada komposisi data latih dengan rata-rata bernilai 87.89%. Lalu referensi dari (Simanjuntak, 2014) yang berjudul “Implementasi *Modified K-Nearest Neighbor* dengan Otomasi Nilai K pada Pengklasifikasian Penyakit Tanaman Kedelai” dalam penelitian tersebut nilai dari K pada algoritme *Modified K-Nearest Neighbor* dapat secara otomatis ditentukan oleh sistem dengan menggunakan metode *Brute Force* untuk mencari nilai K yang paling terbaik, hasil akurasi dari implementasi mencapai rata-rata bernilai 98,83%.

Berdasarkan hasil dari penelitian tersebut, penulis akan melakukan penelitian yang berjudul “Penentuan Penerimaan Beasiswa Menggunakan Metode *Modified K-Nearest Neighbor*”. Metode *Modified K-Nearest Neighbor* menambahkan perhitungan validitas dan pembobotan *weight voting* dari proses K-NN. Digunakan untuk pengklasifikasian data baru yang kelasnya belum diketahui berdasarkan pada nilai *k* tetangga terdekat. Yang pada penelitian sebelumnya terbukti metode MKNN menghasilkan akurasi yang cukup baik.

1.2 Rumusan masalah

Berdasarkan latar belakang, maka rumusan masalah adalah sebagai berikut:

1. Bagaimana model implementasi yang diterapkan untuk metode *Modified K-Nearest Neighbor* pada klasifikasi penentuan penerimaan beasiswa.
2. Berapa tingkat akurasi yang diberikan dari hasil implementasi metode *Modified K-Nearest Neighbor* pada klasifikasi penentuan penerimaan beasiswa.

1.3 Tujuan

Dari rumusan masalah yang telah diberikan, penelitian yang dilakukan bertujuan untuk:

1. Menerapkan metode *Modified K-Nearest Neighbor* untuk mengklasifikasi penentuan penerimaan beasiswa.
2. Menguji tingkat akurasi dari hasil perhitungan metode *Modified K-Nearest Neighbor* untuk mengklasifikasi penentuan pemberian beasiswa.

1.4 Manfaat

Manfaat dari penelitian ini adalah untuk membantu memudahkan petugas seleksi beasiswa dalam penentuan penerimaan beasiswa kepada mahasiswa.

1.5 Batasan masalah

Batasan pada penelitian ini untuk melakukan penelitian secara spesifik dan jelas adalah sebagai berikut:

1. Data yang digunakan sebanyak 370 data pengajuan beasiswa pada beasiswa Toyota Astra pada tahun 2015 dan 2016 di ITS.
2. Parameter yang digunakan ada 5 yaitu nilai IPK, tingkat semester, nominal gaji orang tua, jumlah tanggungan saudara dan nominal beban uang kuliah tunggal (UKT) dengan 2 status penerimaan yaitu diterima atau ditolak.
3. Penelitian akan dilakukan dengan mengimplementasikan C# sebagai bahasa pemrograman.

1.6 Sistematika pembahasan

BAB I PENDAHULUAN

Bab ini membahas mengenai alasan utama dari suatu topik yang diangkat sebagai objek penelitian yang dituangkan dalam latar belakang masalah, identifikasi masalah mengenai poin-poin permasalahan yang akan diselesaikan, rumusan masalah pada penelitian, tujuan penelitian, dan manfaat yang dalam dilaksanakannya penelitian, batasan masalah yang akan dibahas pada penelitian dan sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Membahas mengenai teori-teori dasar dan referensinya sebagai pernyataan pendukung dilakukannya penelitian. Teori yang dibahas pada kajian pustaka mengenai penelitian sebelumnya, pengertian dari klasifikasi dan metode yang digunakan pada penelitian.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan tentang metode penelitian yang digunakan dengan pendekatan penelitian baik dari implementasi, eksperimen, desain, kebutuhan, analisis dan pengujian.

BAB IV PERANCANGAN SISTEM

Bab ini menjelaskan tentang seluruh perancangan sistem yang akan diterapkan pada penelitian dalam klasifikasi untuk penentuan pemberian beasiswa bidik misi kepada mahasiswa.

BAB V IMPLEMENTASI SISTEM

Pada bab ini akan menjelaskan mengenai implementasi dari metode yang diterapkan, pada bab ini akan dijelaskan tahapan dari implementasi metode *Modified K-Nearest Neighbor* dalam memberikan solusi untuk klasifikasi penentuan pemberian beasiswa kepada mahasiswa.

BAB VI PENGUJIAN DAN ANALISIS

Pada bagian ini dilakukan pengujian terhadap sistem yang telah dibuat untuk mengetahui tingkat keberhasilan sistem dalam memberikan suatu solusi. Analisis terhadap hasil pengolahan data baik dari masukan maupun proses yang berjalan pada sistem hingga hasil keluaran sistem.

BAB VII KESIMPULAN DAN SARAN

Setiap proses maupun hasil yang didapatkan selama penelitian akan dibahas secara ringkas pada bab kesimpulan. Saran yang diperoleh dari hasil penelitian dicantumkan dengan harapan akan ada penelitian yang jauh lebih baik untuk dilakukan dari penelitian sebelumnya.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bab landasan kepuasaan disajikan beberapa teori pendukung untuk penelitian “Penentuan Penerimaan Beasiswa Menggunakan Metode *Modified K-Nearest Neighbor*”. Teori yang akan dibahas mengenai metode *Modified K-Nearest Neighbor* dalam Penerimaan Beasiswa.

2.1 Kajian pustaka

Pada sub-bab kajian pustaka ini akan menyajikan penelitian terdahulu dalam menyelesaikan masalah dengan menggunakan metode *Modified K-Nearest Neighbor*. Sebagai acuan dari penelitian ini, telah dilakukan penelitian sebelumnya dengan judul “Diagnosa Penyakit Kulit Pada Kucing Menggunakan Metode *Modified K-Nearest Neighbor*”. Pada penelitian dilakukan dengan tujuan untuk dapat mengetahui penyakit kulit apa yang sedang dialami seekor kucing. Data diperoleh dari klinik hewan *Purple Shop* yang kemudian data tersebut diproses berdasar pada data training dengan perhitungan algoritme untuk memberikan sejumlah kelas klasifikasi sesuai dengan penyakit kulit kucing. Implementasi yang dilakukan dengan tampilan berbasis *web*. Dari penelitian tersebut yang dilakukan, didapatkan sistem klasifikasi yang sesuai dengan permasalahan, dan juga sistem memiliki nilai akurasi yang cukup tinggi (Putri, 2017).

Penelitian selanjutnya yang berjudul “Implementasi *Modified K-Nearest Neighbor* dengan Otomasi Nilai K pada Pengklasifikasian Penyakit Tanaman Kedelai”, nilai dari K terbaik ditentukan dengan *brute force*. Penelitian tersebut menggunakan 266 data dari *Soybean Disease Data Set*, dari penelitian dinyatakan bahwa nilai dari K sangat mempengaruhi nilai dari akurasi klasifikasi, dengan nilai rata rata akurasi dari percobaan sebesar 98,83% (Simanjuntak, 2017). Kemudian dari penelitian yang berjudul “Perbandingan Klasifikasi Beasiswa Toyota Astra menggunakan K-Nearest Neighbor dan Naïve bayes Sebagai Penentu Metode Klasifikasi pada sistem Pendukung Keputusan Penerimaan Beasiswa Toyota Astra” mengenai klasifikasi beasiswa yang membandingkan antara metode KNN dan metode *Naïve Bayes* dimana didapatkan hasil bahwa penggunaan metode *K-Nearest Neighbor* memiliki nilai hasil akurasi yang lebih baik dibandingkan dengan metode *Naïve Bayes* (Lestari, 2017).

Tabel 2.1 Kajian Pustaka

Judul	Objek	Metode	Keluaran
	Masukan dan Parameter	Proses	Hasil Penelitian
Implementasi <i>Modified K-Nearest Neighbor</i> Dengan Otomasi Nilai K Pada Pengklasifikasian	Data dari <i>Center for machine learning and intelligent Syses</i>	<i>Modified K-Nearest Neighbor</i>	- Nilai akurasi klasifikasi - Nilai K paling optimal
	- Jumlah data latih 266 - Jumlah dataset 15 jenis penyakit	- Otomasi Nilai K - Normalisasi - Jarak <i>euclidean</i> - Validitas data	nilai k dapat diotomasi dengan menggunakan <i>brute force</i> dan nilai k yang paling optimal mempengaruhi tingkat akurasi

Judul	Objek	Metode	Keluaran
	Masukan dan Parameter	Proses	Hasil Penelitian
Penyakit Tanaman Kedelai (Paper 1)		- <i>Weight voting</i>	
Diagnosa Penyakit Kulit Pada Kucing menggunakan Metode <i>Modified K-nearest Neighbor</i> (Paper 2)	<i>Dataset</i> penyakit kulit kucing dari klinik hewan Purple Shop	<i>Modified K-Nearest Neighbor</i>	- Pengaruh nilai K - Pengaruh sebaran data - Pengaruh jumlah data latih - Nilai akurasi
	- 15 parameter - 5 jenis penyakit - 240 data	- Normalisasi - Jarak <i>euclidean</i> - Validitas data - <i>Weight voting</i>	<i>Modified K-Nearest Neighbor</i> yang diaplikasikan ke dalam diagnosa penyakit kulit kucing memiliki nilai akurasi yang cukup tinggi.
Perbandingan Klasifikasi Beasiswa Toyota Astra menggunakan <i>K-Nearest Neighbor</i> dan <i>Naïve Bayes</i> sebagai Penentu metode Klasifikasi pada Sistem Pendukung Keputusan Penerimaan Beasiswa (Paper 3)	Data didapatkan dari penerimaan beasiswa Toyota Astra di ITS pada tahun 2015 dan 2016	- <i>K-Nearest Neighbor</i> - <i>Naïve Bayes</i>	- Perbandingan nilai akurasi - Perbandingan Nilai <i>Recall</i> - Perbandingan nilai <i>Presisi</i>
	- 5 Parameter Beasiswa - 2 status penerimaan - 370 data	- Normalisasi - Jarak <i>Euclidean</i> - Mengurutkan jarak terdekat - Klasifikasi <i>Naïve Bayes</i>	Implementasi dari klasifikasi penerimaan beasiswa menggunakan <i>K-Nearest Neighbor</i> lebih baik dari <i>Naïve Bayes</i> berdasarkan nilai akurasi, <i>presisi</i> , <i>recall</i> , sensitivitas dan spesifitas.

2.2 Data Mining

Untuk dapat menemukan pengetahuan yang tersembunyi dari database adalah merupakan kegunaan dari istilah *data mining*. Tujuan utamanya adalah untuk dapat melakukan ekstrak sebuah informasi dari sebuah dataset yang besar dengan mengubahnya kedalam format yang dapat dimengerti dan dipahami dalam penggunaan selanjutnya (V & Ravikumar, 2014).

Meningkatkan pengambilan keputusan dalam menemukan pola data yang kompleks juga merupakan kegunaan dari *data mining*. Dengan tujuan utamanya adalah untuk kumpulan *dataset* diprediksi variabelnya dan mendeskripsikan pola yang ditemukan berupa gambaran data sehingga manusia dapat menginterpretasikan (Oswal, et al., 2014).

2.3 Normalisasi data

Normalisasi data diperlukan dalam alat preprocessing data yang digunakan dalam sistem data mining dan perhitungan untuk mempersempit *range* data latih dan persebaran data merata, dalam normalisasi terdapat beberapa teknik normalisasi seperti contoh *min-max normalization*, *z-score normalization*, *decimal scalling* dan *sigmoidal normalization*. Sedangkan pada penelitian ini normalisasi data yang digunakan adalah *min-max normalization*. *Min-max normalization* adalah sebuah transformasi nilai dari data yang telah ada dengan *range value* terkecil (*min*) sebesar 0 dan nilai terbesar (*max*) sebesar 1, pada persamaan dari *Min-Max Normalization* ditunjukkan pada Persamaan (2.1) (Jain, 2011).

$$V' = \frac{V - \min A}{\max A - \min A} \cdot (new_{\max A} - new_{\min A}) + new_{\min A} \quad (2.1)$$

V' = Nilai dari data baru hasil dari normalisasi

V = Nilai dari data sebelum dinormalisasi

$new_{\max A}$ = Batas nilai maksimum terbaru

$new_{\min A}$ = Batas nilai minimum terbaru

Max A = nilai maksimum pada kolom

Min A = nilai minimum pada kolom

2.4 Algoritme K-Nearest Neighbor

Metode *K-Nearest Neighbor* merupakan salah satu teknik algoritme yang paling sederhana dan paling intuitif dalam bidang diskriminasi statistic. Proses dari perhitungan jarak *Euclidean* dalam algoritme ini adalah pertama membuat dataset dari data latih yang kelasnya telah diketahui terlebih dahulu, langkah selanjutnya adalah *dataset* dari data uji akan diklasifikasikan berdasar pada jarak yang paling dekat dari tiap data latih yang bergantung pada nilai k . Persamaan *Euclidean* untuk menghitung jarak antara titik data latih (x) dengan titik data uji (y) dimana I merupakan representasi nilai atribut dan n dimensi atribut ditunjukkan pada persamaan (2.2) (Hechenbichler, 2004).

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.2)$$

$d(x, y)$ = Jarak *Euclidean* antara titik data latih x dengan data uji y

x_i = Sampel data latih

y_i = Data uji

n = Dimensi atribut

2.5 Algoritme Modified K-Nearest Neighbor

Metode *Modified K-Nearest Neighbor* atau yang biasa disingkat dengan MKNN ini merupakan hasil dari sebuah modifikasi dari metode *K-Nearest Neighbor*, didalam metode MKNN terbagi kedalam dua proses, yaitu yang pertama adalah validasi data latih lalu yang kedua adalah proses pembobotan *K-Nearest Neighbor* atau *weight voting*. Klasifikasi dilakukan pada data uji berdasarkan bobot nilai tertinggi kelas pada k data latih yang telah divalidasi dengan jarak yang terdekat, berbeda dengan metode K-NN yang tidak melalui proses validasi data latih. Metode validasi data latih ini dapat memaksimalkan data latih dengan validitas tinggi dan memiliki jarak yang dekat dengan data uji (Parvin, et al., 2008).

2.5.1 Validitas data

Dalam penggunaan *Modified K-Nearest Neighbor* semua data latih diwajibkan melalui proses validasi terlebih dahulu, validitas setiap titik dihitung menurut tetangga terdekatnya berdasarkan k . Validitas data latih ini adalah berfungsi untuk menghitung jumlah titik dengan tabel yang sama untuk semua data yang ada pada data latih, pada proses ini semua nilai fungsi data s bergantung kepada tetangga terdekatnya bernilai sama atau tidak pada data latih. Persamaan dari validitas data ditunjukkan pada persamaan (2.3) (Parvin, et al., 2008).

$$Validitas(x) = \frac{1}{k} \sum_{i=1}^k S(Label(x))(Ni(x))) \quad (2.3)$$

K = Jumlah titik terdekat

$Label(x)$ = label kelas x

$Ni(x)$ = Label kelas titik terdekat x

S = bernilai 1 apabila kelas sama atau bernilai 0 apabila kelasnya tidak sama

2.5.2 Weight voting

Weight voting adalah merupakan tahapan yang ada pada metode MKNN sebagai bentuk variasi dari metode KNN. Dalam MKNN, setiap data dihitung bobotnya. Teknik ini akan sangat berpengaruh pada data yang memiliki nilai validitas lebih tinggi dan lebih dekat dengan data. Dengan pertama menghitung bobot setiap tetangga menggunakan $1/(d_e+0.5)$. Lalu, validitas sampel data latih dikalikan dengan bobot Euclidean tersebut. mengalikan validitas dengan jarak *Euclidean* dapat mengatasi kelemahan pada setiap data yang memiliki jarak dengan *weight* yang bermasalah dalam *outlier*. Persamaan *weight voting* ditunjukkan pada persamaan (2.4) (Parvin, et al., 2008).

$$W(i) = Validitas(i) \cdot \frac{1}{d_e + \alpha} \quad (2.4)$$

$W(i)$ = Perhitungan Weight Voting ke i

$Validitas(i)$ = nilai validitas ke i

D_e = Jarak *euclidean* data latih dan data uji

$\alpha = 0,5$

2.6 Akurasi sistem

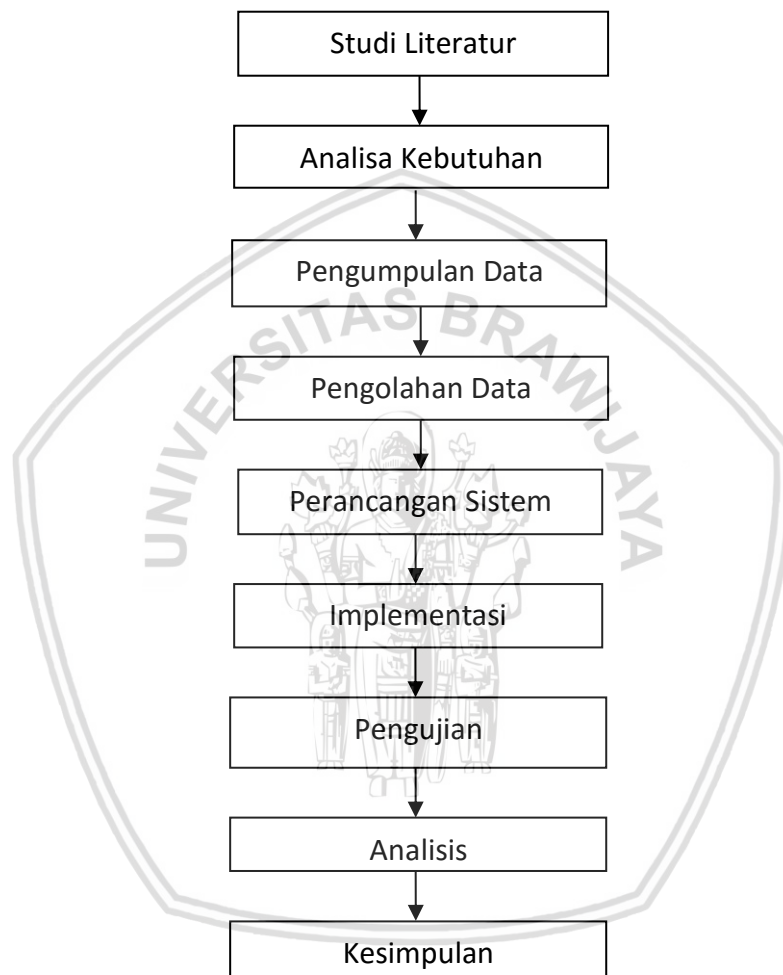
Akurasi sistem adalah merupakan sebuah persamaan yang berfungsi untuk mengetahui seberapa mendekati suatu angka hasil pengkuruan dari sistem terhadap angka hasil sebenarnya. Persamaan ini dihitung dengan cara membagi jumlah data uji yang telah diklasifikasikan dengan benar oleh sistem dengan total data uji dikalikan 100%. Persamaan akurasi ditunjukkan pada persamaan (2.5).

$$Akurasi = \frac{Jumlah\ data\ uji\ benar}{Jumlah\ data} \times 100\% \quad (2.5)$$



BAB 3 METODOLOGI

Pada bab metodologi penelitian dilakukan beberapa tahapan untuk menyelesaikan penelitian “Penentuan Penerimaan Beasiswa Menggunakan Metode *Modified K-Nearest Neighbor*” yang diilustrasikan pada diagram seperti pada Gambar 3.1:



Gambar 3.1 Diagram alir metodologi

3.1 Studi literatur

Pada studi literatur membahas mengenai dasar mengenai keilmuan terkait dan pemahaman terhadap teori-teori yang dilakukan sebagai pendukung penelitian. Sumber dari teori-teori yang digunakan didapatkan dari buku, jurnal, dan penelitian sebelumnya yang telah dilakukan, dan sumber-sumber pustaka yang dapat dipertanggung jawabkan. Sumber terkait dapat membantu dalam penyelesaian dalam penelitian “Penentuan Penerimaan Beasiswa Menggunakan

Metode *Modified K-Nearest Neighbor*". Teori-teori yang mendukung penelitian ini, adalah:

1. Pengetahuan Dasar mengenai *data mining*.
2. Pengetahuan dan implementasi mengenai *Modified K-Nearest Neighbor*.

3.2 Analisis kebutuhan sistem

Analisis kebutuhan sistem dilakukan dalam upaya untuk mengetahui secara spesifik atas apa saja yang dibutuhkan oleh sistem. Kebutuhan secara spesifik dilakukan baik pada kebutuhan *software*, *hardware*, dan kebutuhan data dalam proses penelitian. dilakukannya analisis kebutuhan bertujuan untuk mendukung pembuatan sistem. Kebutuhan yang diperlukan oleh sistem diidentifikasi sebagai berikut:

3.2.1 Kebutuhan Fungsional:

Merupakan spesifikasi sistem dalam mengolah dan menampilkan data hasil dari proses klasifikasi penerimaan beasiswa. Selain itu, kebutuhan sistem untuk menggunakan metode *modified K-Nearest Neighbor*.

3.2.2 Kebutuhan Non-Fungsional:

Pada sistem ini memerlukan spesifikasi kebutuhan secara non-fungsional agar mengetahui secara detail kebutuh sistem baik dari kebutuhan perangkat keras maupun dari kebutuhan perangkat lunak untuk sistem. Rincian kebutuhan untuk sistem secara spesifik adalah:

- a. Kebutuhan perangkat keras
 - 1) Komputer / Laptop dengan OS core i3
 - 2) RAM 4GB
 - 3) ROM 500 GB
- b. Kebutuhan perangkat lunak
 - 1) Berjalan menggunakan Sistem Operasi Windows 10
 - 2) *Visual Studio* 2015

3.3 Pengumpulan data

Pada tahapan ini dilakukan proses pengumpulan data terkait dengan penerimaan beasiswa, data yang digunakan merupakan data mahasiswa penerima beasiswa Toyota Astra, sebanyak 370 data. Parameter yang digunakan dalam penentuan penerimaan beasiswa ada 5 yaitu nilai IPK, tingkat semester, nominal gaji orang tua, jumlah tanggungan saudara, nominal beban uang kuliah tunggal (UKT) dengan 2 status dalam penerimaan beasiswa yaitu diterima atau ditolak. Data tersebut didapat berdasarkan pemberian beasiswa Toyota Astra pada Institut Teknologi Sepuluh Nopember tahun 2015 dan 2016 pada penelitian yang

dilakukan oleh Ditta Resty Dwi Lestari (Lestari, 2017), yang didapatkan dari internet pada repository ITS.

3.4 Pengolahan data

Dari data yang telah dikumpulkan kemudian sekumpulan data tersebut akan diolah agar menjadi informasi yang bermanfaat dengan klasifikasi penerimaan beasiswa menggunakan metode *Modified K-Nearest Neighbor*. Hasil yang di proses dengan dilakukannya normalisasi data Lalu akan dilakukan klasifikasi dengan metode KNN dengan menambahkan perhitungan validitas dan *weight voting*.

3.5 Perancangan sistem

Deskripsi mengenai perancangan sistem akan dibahas pada tahap ini. Perancangan yang diberikan berupa manajemen data, perancangan algoritme, manualisasi algoritme, serta proses yang digambarkan melalui suatu blok diagram perancangan sistem pada bab IV.

3.6 Implementasi sistem

Sistem akan diimplemetasikan secara runtut dengan menggunakan bahasa C# yang dibantu dengan software *Visual Studio*. Implementasi yang mengacu pada program ini dengan perancangan sistem yang menerapkan algoritme dengan menggunakan pendekatan berdasar data latih. Dilakukan pula perhitungan manual pada sebagian data untuk klasifikasi status penerimaan beasiswa.

3.7 Pengujian

Pengujian yang dilakukan bertujuan dalam memberikan informasi akurat mengenai tingkat akurasi algoritme yang dilakukan oleh sistem. Algoritme tersebut akan memberikan suatu penilaian terhadap kualitas klasifikasi pada aplikasi yang dijalankan. Selain pengujian yang dilakukan pada algoritme klasifikasi, akan dilakukan pengujian dengan menggunakan pengujian pengaruh nilai K, pegujian kelas seimbang tidak seimbang dan pengujian pengaruh jumlah data latih yang menjadi indikator keberhasilan *fungsiionalitas sistem*.

3.8 Analisis

Untuk analisis sistem akan dilakukan perbandingan antara data sebelum diolah dan setelah data diolah oleh sistem. Dari analisa sistem tersebut didapatkan suatu informasi mengenai dapat mengklasifikasi pemberian beasiswa kepada mahasiswa yang tepat. Selain itu Analisa dilakukan untuk menentukan nilai akurasi yang telah diberikan oleh perhitungan sistem dengan perhitungan secara manual.

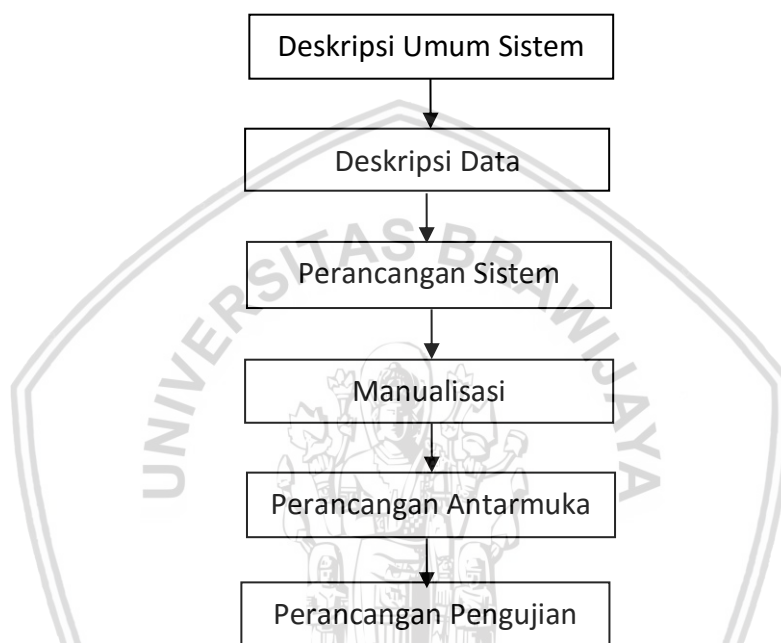
3.9 Kesimpulan

Untuk mendapatkan suatu kesimpulan maka perlu diselesaikan seluruh tahapan perancangan dan analisis metode. Agar penelitian dapat berkembang, maka dibutuhkan saran sehingga kekurangan yang ada dapat diperbaiki dan menjadi suatu pertimbangan jika dilakukan penelitian yang bertujuan untuk mengembangkan sistem maupun implementasi metode.



BAB 4 PERANCANGAN SISTEM

Bab ini berisi analisis perancangan perangkat lunak yang dibutuhkan untuk merancang sebuah sistem agar berguna untuk sebaik mungkin dapat membantu petugas dalam penentuan mahasiswa penerima beasiswa dengan menggunakan metode *Modified K-Nearest Neighbor*. Bab perancangan meliputi deskripsi umum sistem, deskripsi data, perancangan sistem, perhitungan manual, perancangan antarmuka. Pada Gambar 4.1 ditunjukkan alur proses yang akan dilakukan pada bab ini.



Gambar 4.1 Diagram alir perancangan sistem

4.1 Deskripsi umum sistem

Pada sistem yang akan dikembangkan ini adalah merupakan sebuah sistem untuk penentuan penerimaan beasiswa mahasiswa yang berdasarkan pada data mahasiswa ketika mengajukan mendapatkan beasiswa Toyota astra. Metode yang digunakan pada sistem adalah metode *Modified k-Nearest Neighbor* (MKNN) menggunakan dataset data pengajuan beasiswa mahasiswa dengan lima fitur yaitu diantaranya nilai IPK, tingkat semester, nominal gaji orang tua, jumlah tanggungan saudara, nominal beban uang kuliah tunggal (UKT) dengan dua status hasil yaitu diterima dan ditolak.

Sebagai proses awal dari sistem, sistem akan melakukan pembacaan data yang tersimpan dalam file *excel* dengan *extension .xls* yang telah terbagi antara dua *worksheet* yaitu data latih dan data uji untuk menyimpannya sementara pada *memory* sistem sebelum digunakan untuk proses selanjutnya. Proses selanjutnya sistem akan melakukan *normalisasi* data dari data yang telah dilakukan pembacaan pada proses awal. Kemudian sistem akan menghitung jarak antar data yang akan menjadi masukan dalam menghitung nilai validitas dari setiap data latih.

Nilai dari *validitas* pada setiap data *training* diperoleh dari input jarak tiap data latih dan nilai k .

Proses berikutnya yaitu menghitung jarak dari data latih dan data uji yang nilainya berguna untuk masukan dalam menghitung pembobotan dengan *weight voting*. Kemudian berdasarkan nilai *weight voting* yang dihasilkan akan menentukan status diterima atau ditolak pada proses klasifikasi penentuan penerimaan beasiswa mahasiswa kemudian akan dilakukan perhitungan berapa hasil tingkat akurasi dari sistem.

4.2 Deskripsi data

Dataset yang digunakan dalam penelitian ini adalah Data yang digunakan pada penelitian ini yaitu berupa data hasil rekap pemberian beasiswa dari Toyota astra pada tahun 2015 dan 2016 di Institut Teknologi Sepuluh Nopember sebanyak 370 data yang didapat dari penelitian sebelumnya yang dilakukan oleh Ditta Resty Dwi Lestari (Lestari, D.,2017). Parameter yang digunakan dalam penentuan penerimaan beasiswa ada 5 yaitu nilai IPK, tingkat semester, nominal gaji orang tua, jumlah tanggungan saudara, nominal beban uang kuliah tunggal (UKT) dengan 2 status dalam penerimaan beasiswa yaitu diterima atau ditolak. Dari 370 mahasiswa yang mengajukan beasiswa tersebut sebanyak 46 mahasiswa diterima untuk mendapat beasiswa, sedangkan sebanyak 324 mahasiswa ditolak untuk mendapat beasiswa. Parameter beserta kode parameter ditunjukkan pada Tabel 4.1.

Tabel 4.1 Parameter dan kode parameter

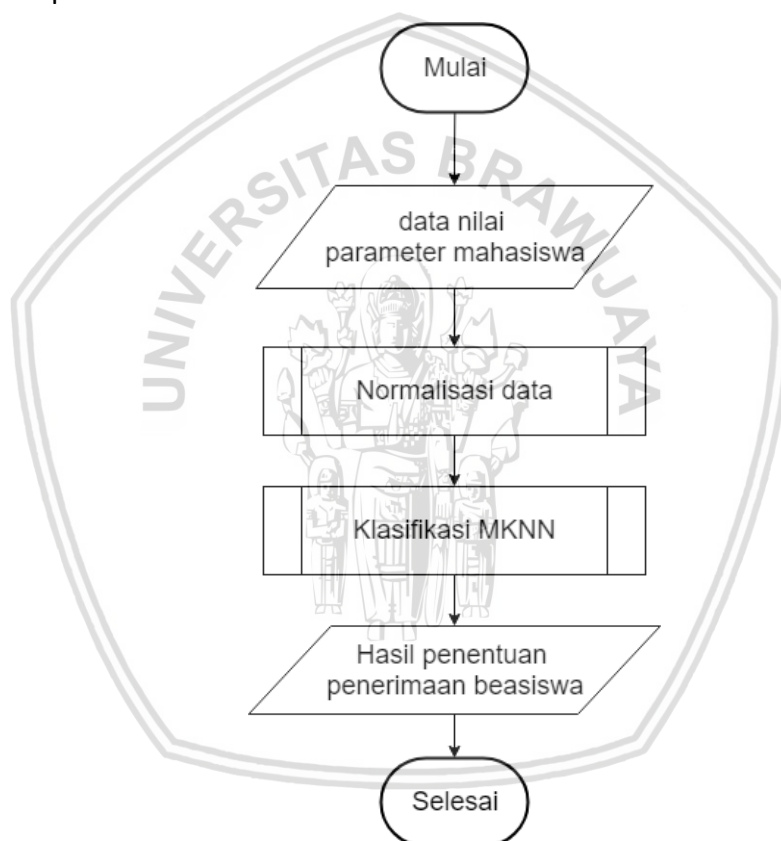
No	Nama Parameter	Kode Parameter
1	Nilai IPK mahasiswa	P1
2	Tingkat semester mahasiswa	P2
3	Nominal gaji orang tua mahasiswa	P3
4	Jumlah tanggungan saudara	P4
5	Nominal beban uang kuliah tunggal (UKT)	P5

4.3 Perancangan sistem

Pada bagian perancangan sistem ini akan dijelaskan mengenai proses-proses yang dibutuhkan untuk membangun sebuah sistem agar dapat memenuhi kebutuhan petugas seleksi beasiswa, sistem yang akan dibangun ini merupakan sistem penentuan penerimaan beasiswa dengan menggunakan metode *Modified K-Nearest Neighbor*. Sistem akan menerima masukan dari pengguna sistem berupa data parameter mahasiswa, yang kemudian data tersebut akan diolah oleh sistem sehingga menghasilkan keluaran berupa hasil penerimaan beasiswa mahasiswa.

4.3.1 Perancangan proses

Proses awal sistem dalam memberikan hasil penentuan penerimaan beasiswa dengan memasukkan data berupa data nilai dari parameter mahasiswa yang kemudian akan dilakukan normalisasi data yang berfungsi untuk mempersempit range data. Setelah dilakukannya normalisasi data selanjutnya sistem akan memulai proses klasifikasi MKNN dengan menghitung jarak *euclidean* antar data latih. Setelah hasil *euclidean* telah didapat maka dilanjutkan dengan melakukan perhitungan validitas data latih berdasar dari nilai *euclidean* data latih. Proses selanjutnya yang terakhir adalah melakukan perhitungan *weight voting* yang dapat digunakan oleh sistem sebagai keluaran hasil penentuan pengajuan beasiswa mahasiswa tersebut diterima atau ditolak. Diagram alir proses sistem ditunjukkan pada Gambar 4.2.

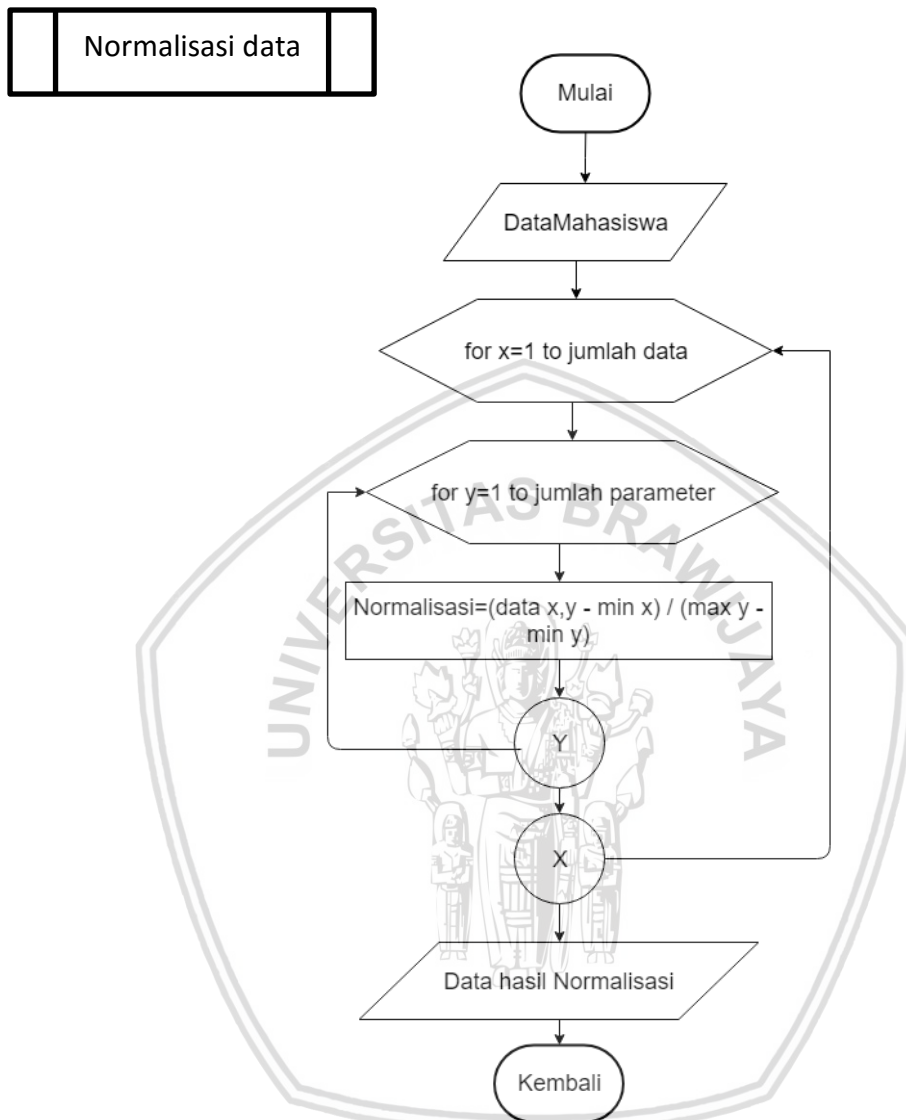


Gambar 4.2 Diagram alir proses sistem

4.3.2 Proses normalisasi data

Data yang akan digunakan dalam proses klasifikasi menggunakan *Modified K-Nearest Neighbor* terlebih dahulu harus dinormalisasi yang bertujuan untuk range data berada pada *range* [0,1] sehingga sebaran dari data tersebut tidak terlalu jauh, data yang akan digunakan pada penelitian ini adalah nilai parameter dari mahasiswa. Pertama proses perhitungan normalisasi dimulai saat *user* memasukkan data nilai parameter mahasiswa yang kemudian oleh sistem akan dilakukan perhitungan sejumlah data gejala yang telah dimasukkan sebelumnya, kemudian melakukan perhitungan normalisasi. Dalam melakukan perhitungan

normalisasi ini menggunakan Persamaan (2.1) dimana menghasilkan keluaran sebanyak data yang dimasukkan. Diagram alir normalisasi data ditunjukkan pada Gambar 4.3.

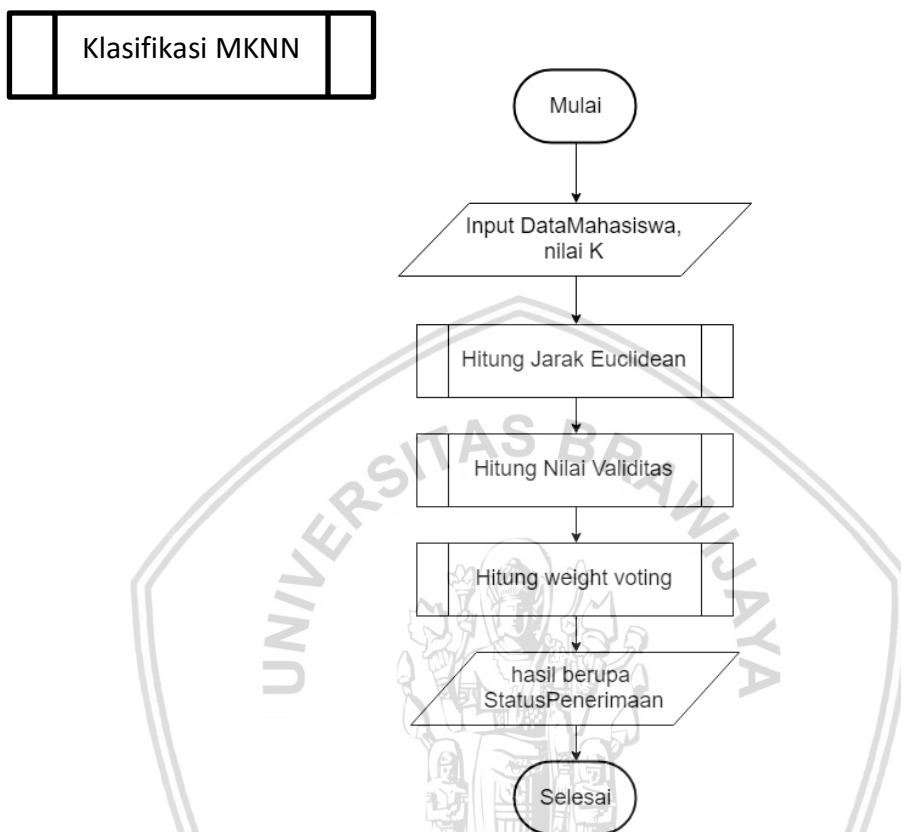


Gambar 4.3 Diagram alir normalisasi data

4.3.3 Proses klasifikasi MKNN

Pada proses klasifikasi Modified K-Nearest Neighbor ini berfungsi sebagai pendukung pengambilan keputusan yang berdasarkan dari hasil perhitungan dalam menentukan status penerimaan beasiswa mahasiswa. Proses perhitungan yang ada dalam *Modified K-Nearest Neighbor* yaitu perhitungan jarak *Euclidean*, perhitungan nilai validitas dan perhitungan *weight voting*. Proses perhitungan jarak *Euclidean* dilakukan pada data latih dalam setiap nilai parameter, kemudian untuk proses selanjutnya yaitu perhitungan nilai validitas data latih berdasarkan nilai jarak *Euclidean* yang sebelumnya telah dihitung, Setelah itu hitung nilai jarak *Euclidean* antara data uji dengan data latih dalam setiap parameter. Pada proses perhitungan yang terakhir dalam *Modified K-Nearest Neighbor* yaitu adalah

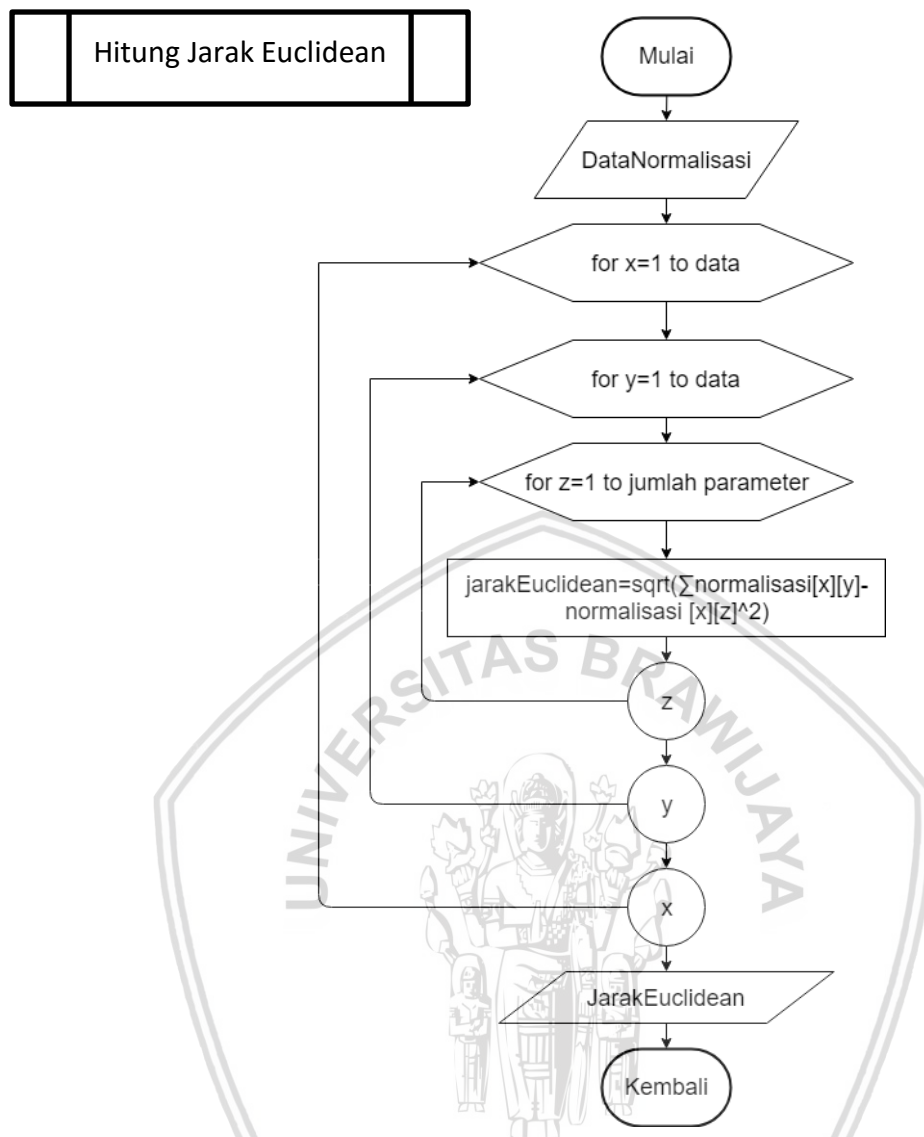
perhitungan *weight voting* yang berdasarkan pada nilai validitas dan nilai jarak *Euclidean* data uji. Setelah melakukan perhitungan *weight voting* lalu sistem akan mengurutkan nilai *weight voting* dari nilai yang paling besar hingga ke nilai yang paling kecil yang menghasilkan berupa hasil penerimaan pengajuan beasiswa mahasiswa. Diagram alir proses klasifikasi MKNN ditunjukkan pada Gambar 4.4.



Gambar 4.4 Diagram alir proses klasifikasi MKNN

4.3.4 Perhitungan jarak Euclidean

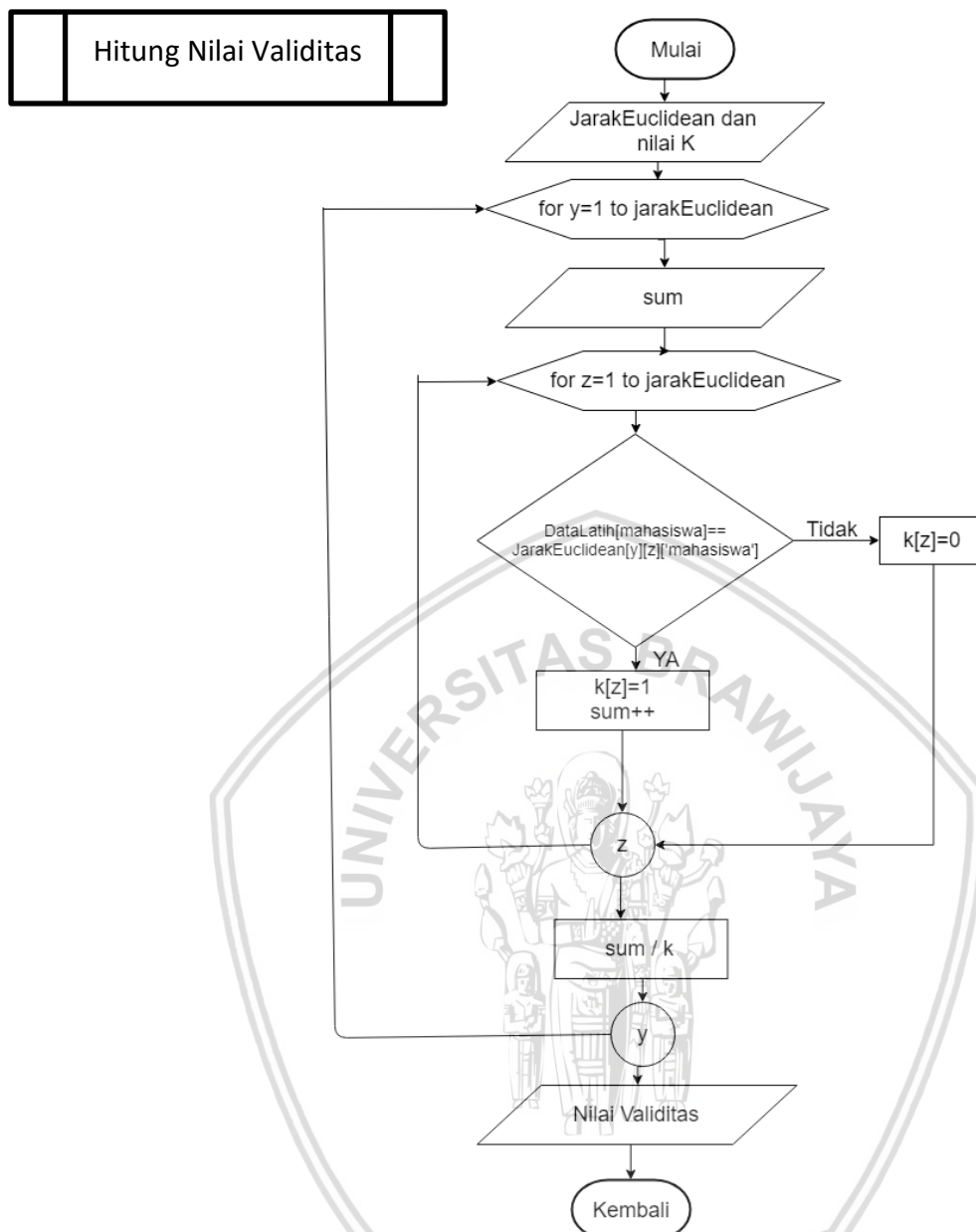
Pada proses perhitungan jarak *Euclidean*, pertama dimulai dengan memasukkan dataset nilai parameter mahasiswa yang nantinya akan dilanjutkan dengan menggunakan proses inisialisasi. Selanjutnya akan digunakan Persamaan (2.2) yang berfungsi untuk melakukan perhitungan jarak *Euclidean* dan menghasilkan nilai jarak *euclidean*. Diagram alir proses perhitungan jarak *euclidean* ditunjukkan pada Gambar 4.5.



Gambar 4.5 Diagram alir Perhitungan Jarak *Euclidean*

4.3.5 Perhitungan validitas

Pada proses perhitungan validitas, pertama dimulai dengan memasukkan data latih data nilai parameter pengajuan beasiswa mahasiswa dan menentukan nilai k. kemudian setelah itu menggunakan persamaan (2.3) untuk melakukan perhitungan validitasnya. Diagram alir proses perhitungan validitas ditunjukkan pada Gambar 4.6.

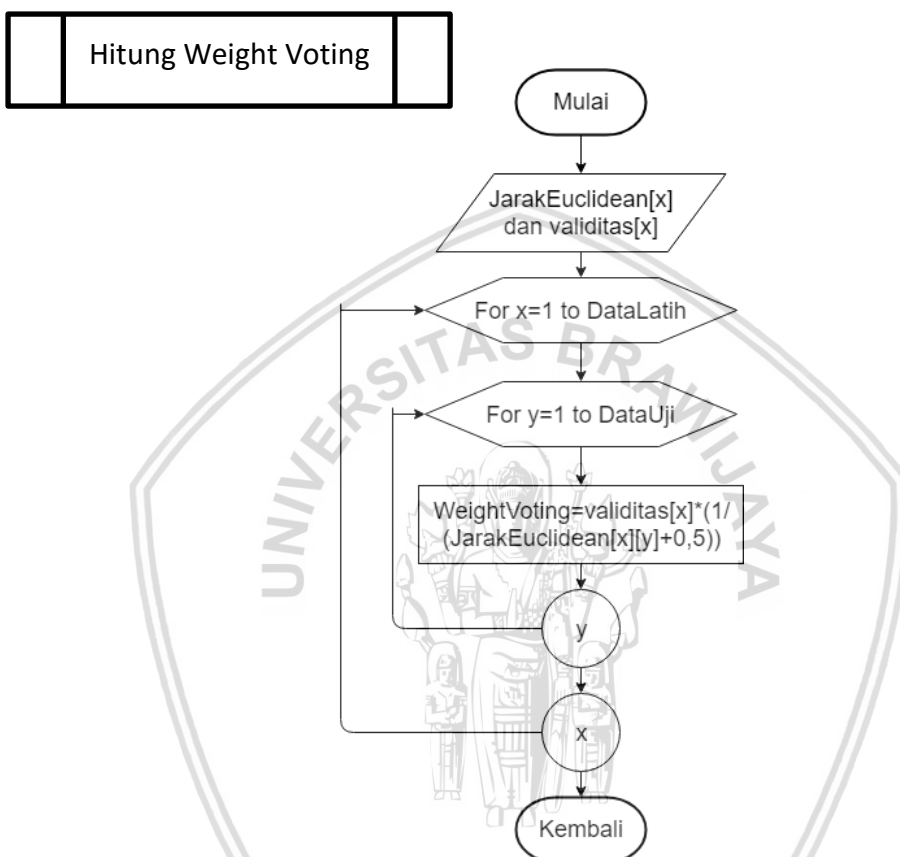


Gambar 4.6 Diagram alir Perhitungan Validitas

Setelah melakukan input data latih nilai parameter pengajuan beasiswa mahasiswa dan menentukan nilai dari k, kemudian sistem akan melakukan perhitungan jarak *euclidean* antar tiap data latih untuk mengetahui jarak terdekat dari setiap tetangga. Sistem akan membandingkan nilai jarak *euclidean* dari setiap data latih dengan setiap kelasnya, dengan persyaratan apabila kelasnya sama maka akan bernilai 1 sedangkan apabila kelasnya tidak sama maka bernilai 0. Setelah didapatkan nilai dari setiap variabel k, maka semua nilai tersebut akan dijumlahkan kemudian dibagi sesuai berdasarkan nilai k yang digunakan dalam mendapatkan nilai *validitas*.

4.3.6 Perhitungan weight voting

Pada proses perhitungan *weight voting* terdapat beberapa tahapan proses yaitu diantaranya adalah memasukkan nilai jarak *Euclidean* data uji dengan data latih dan nilai *validitas* yang sebelumnya telah dihitung, lalu setelah itu sistem akan mulai melakukan perulangan yang kemudian akan melakukan perhitungan *weight voting* dengan menggunakan persamaan (2.4). Diagram alir proses perhitungan *weight voting* ditunjukkan pada Gambar 4.7.



Gambar 4.7 Diagram alir Perhitungan *Weight Voting*

4.4 Manualisasi

Pada penelitian ini, metode yang digunakan dalam penentuan penerimaan beasiswa adalah menggunakan *Modified K-Nearest Neighbor*. Berikut merupakan tahapan yang dilalui pada perhitungan manualisasinya.

1. Menentukan nilai dari k , yang berfungsi untuk mengetahui manakah tetangga terdekatnya. Untuk perhitungan manualisasi ini, nilai dari k adalah $k=3$.
2. Melakukan normalisasi data, agar *range* dari data tidak terlalu jauh yaitu sebesar antara 0 dan 1.
3. Melakukan perhitungan jarak *Euclidean* antar data latih yang berfungsi untuk mengetahui jarak terdekatnya.
4. Melakukan perhitungan nilai validitas antar setiap data latih yang berdasarkan pada proses sebelumnya yaitu nilai jarak *Euclidean*.

5. Melakukan perhitungan nilai jarak *Euclidean* antara data uji dengan data latih.
6. Melakukan perhitungan *weight voting* yang berdasarkan pada proses sebelumnya yaitu nilai jarak *Euclidean* antara data uji dengan data latih dan nilai validitas data.
7. Penentuan kelas yang berdasarkan dari data uji berdasarkan nilai k yang digunakan.

Data yang digunakan untuk perhitungan manualisasi pada penelitian ini adalah data nilai parameter dari syarat pengajuan beasiswa yang terdiri dari 15 dataset seperti pada Tabel 4.2, Dari dataset tersebut akan dipisahkan menjadi 10 data latih dan 5 data uji. Untuk lebih jelasnya, data latih terdapat pada Tabel 4.3 sedangkan untuk data uji terdapat pada Tabel 4.4.

Tabel 4.2 Dataset

No	P1	P2	P3	P4	P5	Status
1	3.27	7	2287900	2	4000000	diterima
2	2.93	7	4469900	5	7500000	ditolak
3	3.2	5	4118300	3	7500000	ditolak
4	3.25	3	2000000	2	1000000	ditolak
5	3.14	3	7500000	2	7500000	ditolak
6	2.88	7	2000000	2	1000000	ditolak
7	2.96	5	2400000	1	2500000	diterima
8	3.04	3	4600000	0	4000000	ditolak
9	3.26	5	4599200	2	2500000	ditolak
10	3.48	7	4445500	3	7500000	diterima
11	3.5	7	2000000	1	4000000	diterima
12	3.63	5	3000000	4	2500000	ditolak
13	3.46	5	4500000	1	4000000	ditolak
14	3.35	5	4500000	2	2500000	ditolak
15	3.15	5	4601200	3	2500000	ditolak

Tabel 4.3 Data Latih

No	P1	P2	P3	P4	P5	Status
1	3.27	7	2287900	2	4000000	diterima
2	2.93	7	4469900	5	7500000	ditolak
3	3.2	5	4118300	3	7500000	ditolak
4	3.25	3	2000000	2	1000000	ditolak
5	3.14	3	7500000	2	7500000	ditolak
6	2.88	7	2000000	2	1000000	ditolak
7	2.96	5	2400000	1	2500000	diterima
8	3.04	3	4600000	0	4000000	ditolak

9	3.26	5	4599200	2	2500000	ditolak
10	3.48	7	4445500	3	7500000	diterima

Tabel 4.4 Data Uji

No	P1	P2	P3	P4	P5	Status
11	3.5	7	2000000	1	4000000	diterima
12	3.63	5	3000000	4	2500000	ditolak
13	3.46	5	4500000	1	4000000	ditolak
14	3.35	5	4500000	2	2500000	ditolak
15	3.15	5	4601200	3	2500000	ditolak

P1= Nilai IPK mahasiswa

P2= Tingkat semester mahasiswa

P3= Nominal gaji orang tua mahasiswa

P4= Jumlah tanggungan saudara

P5= Nominal beban uang kuliah tunggal (UKT)

4.4.1 Menentukan nilai k

Langkah pertama dari perhitungan manualisasi *Modified K-Nearest Neighbor* pada penelitian ini adalah menentukan nilai dari k, yang berfungsi untuk mengetahui manakah tetangga terdekatnya. Untuk perhitungan manualisasi ini, nilai dari k adalah k=3.

4.4.2 Normalisasi data

Setelah menentukan nilai k, tahapan selanjutnya adalah melakukan perhitungan normalisasi data untuk setiap data latih dan data uji yang berfungsi agar *range* dari data tidak terlalu jauh yaitu sebesar antara 0 dan 1. Pada proses ini menggunakan Persamaan (2.1). Hasil dari normalisasi data ditunjukkan pada Tabel 4.5 untuk data latih dan Tabel 4.6 untuk data uji.

a. Normalisasi data pada data latih kolom pertama baris pertama.

$$V' = \frac{3,27 - 2,88}{3,48 - 2,88} (1 - 0) + 0 = 0,65$$

b. Normalisasi data pada data latih kolom pertama baris kedua.

$$V' = \frac{2,93 - 2,88}{3,48 - 2,88} (1 - 0) + 0 = 0,0833$$

Tabel 4.5 Normalisasi Data Latih

No	P1	P2	P3	P4	P5	Status
1	0.65	1	0.052345	0.4	0.461538	diterima
2	0.083333	1	0.449073	1	1	ditolak
3	0.533333	0.5	0.385145	0.6	1	ditolak
4	0.616667	0	0	0.4	0	ditolak
5	0.433333	0	1	0.4	1	ditolak
6	0	1	0	0.4	0	ditolak
7	0.133333	0.5	0.072727	0.2	0.230769	diterima
8	0.266667	0	0.472727	0	0.461538	ditolak
9	0.633333	0.5	0.072727	0.2	0.230769	ditolak
10	1	1	0.444636	0.6	1	diterima

Untuk selanjutnya, melakukan perhitungan normalisasi pada data uji. Sama seperti melakukan perhitungan normalisasi data latih yaitu menggunakan Persamaan (2.1).

- a. Normalisasi data pada data latih kolom pertama baris pertama

$$V' = \frac{3,5 - 3,15}{3,63 - 3,15} (1 - 0) + 0 = 0,729$$

- b. Normalisasi data pada data latih kolom pertama baris kedua

$$V' = \frac{3,63 - 3,15}{3,63 - 3,15} (1 - 0) + 0 = 1$$

Tabel 4.6 Normalisasi Data Uji

No	P1	P2	P3	P4	P5	Status
1	0.729167	1	0	0	1	diterima
2	1	0	0.384438	1	0	ditolak
3	0.645833	0	0.961095	0	1	ditolak
4	0.416667	0	0.961095	0.333333	0	ditolak
5	0	0	1	0.666667	0	ditolak

4.4.3 Perhitungan jarak Euclidean data latih

Tahapan selanjutnya setelah menghitung normalisasi data adalah melakukan perhitungan jarak *Euclidean* data latih, perhitungan ini menggunakan Persamaan (2.2). Hasil dari perhitungan jarak *Euclidean* ditunjukkan pada Tabel 4.7.

- a. Perhitungan jarak *Euclidean* data latih pada kolom pertama baris pertama

$$D(1,1) = \sqrt{(0,65 - 0,65)^2 + (1 - 1)^2 + (0,0523 - 0,0523)^2 + (0,4 - 0,4)^2 + (0,461 - 0,461)^2}$$

$$= 0$$

b. Perhitungan jarak *Euclidean* data latih pada kolom pertama baris kedua

$$D(1,2) = \sqrt{(0,083 - 0,65)^2 + (1 - 1)^2 + (0,449 - 0,0523)^2 + (1 - 0,4)^2 + (1 - 0,461)^2}$$

$$= 1.062$$

Tabel 4.7 Jarak *Euclidean* data latih

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	Status
D1	0	1.062	0.839	1.103	1.494	0.798	0.781	1.218	0.692	0.778	diterima
D2	1.062	0	0.785	1.687	1.336	1.252	1.275	1.524	1.226	1.0001	ditolak
D3	0.839	0.785	0	1.202	0.823	1.312	1.004	0.989	0.805	0.686	ditolak
D4	1.103	1.687	1.202	0	1.426	1.174	0.762	0.847	0.725	1.544	ditolak
D5	1.494	1.336	0.823	1.426	0	1.785	1.353	0.869	1.076	1.292	ditolak
D6	0.798	1.252	1.312	1.174	1.785	0	0.605	1.291	0.963	1.495	ditolak
D7	0.781	1.275	1.004	0.762	1.353	0.605	0	0.721	0.670	1.375	diterima
D8	1.218	1.524	0.989	0.847	0.869	1.291	0.721	0	0.773	1.479	ditolak
D9	0.692	1.226	0.805	0.725	1.076	0.963	0.670	0.773	0	1.008	ditolak
D10	0.778	1.0001	0.686	1.544	1.292	1.495	1.375	1.479	1.008	0	diterima

4.4.4 Validitas data

Tahapan selanjutnya setelah melakukan perhitungan jarak *Euclidean* data latih adalah menghitung nilai dari validitas data. Perhitungan validitas dilakukan sesuai dengan nilai k yang digunakan, dimana nilai k pada perhitungan manualisasi ini adalah k=3, sehingga akan diambil 3 tetangga terdekat dari setiap kolom kemudian melihat kesamaan dari status tetangga terdekat tersebut, apabila statusnya sama maka bernilai 1 sebaliknya apabila statusnya berbeda maka akan bernilai 0 seperti contoh pada kolom D1, 3 nilai terdekatnya adalah 0.586, 0.778 dan 0.781 dimana terdapat 2 kelas status yang sama dari 0.778 dan 0.781. Dalam perhitungan validitas data ini menggunakan Persamaan (2.3). Hasil perhitungan validitas ditunjukkan pada Tabel 4.8.

$$Validitas(1) = \frac{1}{3} \times (0+1+1) = 0,666$$

Tabel 4.8 Validitas Data

No	K1	K2	K3	SUM	Validitas
1	0	1	1	2	0,666
2	1	0	0	1	0,333
3	0	1	1	2	0,666
4	0	1	1	2	0,666

5	1	1	1	3	1
6	0	0	1	1	0,333
7	0	0	0	0	0
8	0	1	1	2	0,666
9	0	0	1	1	0,333
10	0	1	0	1	0,333

4.4.5 Jarak Euclidean data latih dan data uji

Tahapan selanjutnya setelah melakukan perhitungan validitas adalah melakukan perhitungan jarak *Euclidean* antara data uji dan data latih dengan menggunakan persamaan rumus yang sama pada perhitungan jarak *Euclidean* data uji yaitu Persamaan (2.2). Hasil perhitungan jarak *Euclidean* data uji dan data latih ditunjukkan pada Tabel 4.9.

- a. Perhitungan jarak Euclidean data latih dan data uji, baris pertama data uji dan data latih.

$$D(1,1) = \sqrt{(0,729 - 0,65)^2 + (1 - 1)^2 + (0 - 0,052)^2 + (0 - 0,4)^2 + (1 - 0,461)^2}$$

$$= 0,677$$

- b. Perhitungan jarak Euclidean data latih dan data uji, baris pertama data uji dan baris kedua data latih.

$$D(1,2) = \sqrt{(0,729 - 0,083)^2 + (1 - 1)^2 + (0 - 0,449)^2 + (0 - 1)^2 + (1 - 1)^2}$$

$$= 1,272$$

Tabel 4.9 Jarak *Euclidean* data uji dan data latih

No	1	2	3	4	5
1	0.677458	1.343802	1.508574	1.448358	1.613902
2	1.27231	1.686551	1.605794	1.678611	1.556141
3	0.892574	1.275844	0.976924	1.290909	1.384534
4	1.473994	0.809159	1.443802	0.98395	1.204737
5	1.499173	1.435279	0.454608	1.003113	1.122003
6	1.300649	1.583602	1.871045	1.449749	1.439136
7	1.114463	1.338481	1.390531	1.091108	1.182645
8	1.314276	1.326118	0.819887	0.764941	1.003292
9	1.110961	0.897479	1.113785	0.770252	1.026473
10	0.794387	1.470926	1.323696	1.636497	1.82013

4.4.6 Weight voting

Tahapan selanjutnya setelah melakukan perhitungan jarak *Euclidean* antara data latih dan data uji adalah melakukan perhitungan *weight voting*. Dalam perhitungan *weight voting* menggunakan Persamaan (2.4). hasil perhitungan *weight voting* ditunjukkan pada Tabel 4.10.

a. Perhitungan *weight voting* kolom pertama baris pertama

$$W(i) = Validitas(i) \times \frac{1}{d_e + \alpha} = 0,666 \times \frac{1}{0,677+0,5} = 0.565625$$

b. Perhitungan *weight voting* kolom pertama baris kedua

$$W(i) = Validitas(i) \times \frac{1}{d_e + \alpha} = 0,333 \times \frac{1}{1,272+0,5} = 0.18789$$

Tabel 4.10 Weight Voting

No	1	2	3	4	5	Status
1	0.565625	0.36121	0.331579	0.341826	0.315057	diterima
2	0.18789	0.152295	0.158135	0.15285	0.161954	ditolak
3	0.478251	0.375033	0.450937	0.371878	0.353403	ditolak
4	0.337387	0.508724	0.342627	0.448802	0.390676	ditolak
5	0.500207	0.516721	1.047548	0.665286	0.616522	ditolak
6	0.184933	0.159819	0.140444	0.170791	0.171726	ditolak
7	0	0	0	0	0	diterima
8	0.367089	0.364708	0.504589	0.526507	0.443028	ditolak
9	0.206709	0.238286	0.206347	0.262153	0.21815	ditolak
10	0.257265	0.168956	0.182596	0.155863	0.143526	diterima

Setelah mendapatkan nilai dari *weight voting*, langkah selanjutnya yang terakhir adalah menentukan kelas hasil dari perhitungan manual yang didapatkan. Proses awalnya adalah mengambil nilai terbesar dari setiap kolom pada tabel 4.10, dikarenakan nilai k yang digunakan pada perhitungan manualisasi ini bernilai 3 maka hanya diambil 3 nilai terbesar. Jika terdapat data yang sama, maka kemudian data tersebut akan dijumlahkan dan kemudian dibandingkan dengan data lain yang memiliki kelas berbeda. Nilai *weight voting* yang terbesar akan digunakan sebagai prediksi hasil sistem. Seperti contoh pada kolom 2, nilai *weight voting* terbesar terdapat pada data ke-5 dengan nilai 0,516721, terbesar kedua pada data ke-4 sebesar 0,508724 dan terbesar ketiga pada data ke-3 sebesar 0,375033. Ketiga data tersebut berada pada kelas yang sama yaitu dengan status ditolak apabila kelas sama maka dijumlahkan seluruh nilainya yaitu $0,516721+0,508724+0,375033 = 1.400478$, dibandingkan dengan nilai status diterima yang 0, Maka dapat dapat disimpulkan bahwa kelas yang diambil adalah kelas dengan status ditolak. Berikut merupakan hasil penentuan kelas pada Tabel 4.11.

Tabel 4.11 Tabel penentuan kelas hasil

Nilai K	Data uji	Hasil	
		Data Asli	Prediksi sistem
3	1	Diterima	Ditolak
	2	Ditolak	Ditolak
	3	Ditolak	Ditolak
	4	Ditolak	Ditolak
	5	ditolak	Ditolak

4.5 Perancangan antarmuka

Pada perancangan antarmuka sistem ini akan digambarkan secara garis besar bagaimana rancangan tampilan dari sistem yang akan diimplementasikan, antarmuka adalah merupakan alat komunikasi antara pengguna dan sistem penerimaan beasiswa, agar dapat mudah dipahami oleh pengguna maka antarmuka akan didesain sesederhana mungkin. Pada perancangan antarmuka sistem ini akan menampilkan rancangan halaman utama sistem dan halaman hasil penerimaan beasiswa.

4.5.1 Antarmuka halaman utama

Pada halaman utama dari sistem adalah merupakan tampilan awal dari sistem penerimaan beasiswa mahasiswa. Pada halaman utama ini terdapat dua tombol *choose file* yang berfungsi agar *user* dapat memasukkan data latih dan data uji dengan dengan format .xls yang berada pada satu *Excel* namun berbeda *sheet*. Kemudian setelah itu agar *user* dapat memasukkan nilai k dibuat *textbox* sehingga *user* dapat memasukkan nilai k pada *textbox* tersebut. Setelah itu terdapat tombol inialisasi yang berfungsi untuk menampilkan masukkan data latih dan data uji yang selanjutnya digunakan untuk melakukan perhitungan, hasil perhitungan menggunakan metode *Modified K-Nearest Neighbor*. Tampilan antarmuka halaman utama ditunjukkan pada Gambar 4.8.

Gambar 4.8 Perancangan antarmuka halaman utama

4.5.2 Antarmuka halaman hasil penerimaan beasiswa

Pada halaman hasil penerimaan beasiswa ini, user dapat melihat hasil klasifikasi dari pemrosesan yang dilakukan oleh sistem menggunakan metode *Modified K-Nearest Neighbor* berupa hasil status penerimaan beasiswa dan hasil akurasi. Terdapat tombol yang berfungsi untuk dapat melihat proses sistem seperti melihat normalisasi data latih, normalisasi data uji, jarak *Euclidean*, validitas dan *weight voting* pada tab diantara tab data dan tab hasil. Tampilan antarmuka halaman hasil penerimaan beasiswa ditunjukkan pada Gambar 4.9.

Gambar 4.9 Perancangan antarmuka halaman hasil

4.6 Perancangan Pengujian

Sebagai proses pengujian untuk mengetahui tingkat akurasi dari hasil keluaran sistem dalam penelitian ini, terdapat tiga jenis proses pengujian yaitu pengujian terhadap nilai k , pengujian pengaruh sebaran data antara pada data latih seimbang dan data latih tidak seimbang, pengujian pengaruh jumlah data latih dengan jumlah data uji tetap. Data sampel parameter pengajuan beasiswa mahasiswa diambil secara acak.

4.6.1 Rancangan pengujian pengaruh nilai k

Pada rancangan pengujian ini berfungsi untuk mengetahui pengaruh nilai k yang berbeda terhadap nilai akurasi dengan menggunakan jumlah data latih yang telah ditentukan, pengujian dilakukan sebanyak n kali. Rancangan pengujian pengaruh nilai k ditunjukkan pada Tabel 4.12.

Tabel 4.12 Rancangan Pengujian Pengaruh Nilai K

Nilai K	Percobaan			Rata-rata Akurasi (%)
	Data Latih 1	Data Latih 2	Data Latih 3	
1				
2				
...				
n				

4.6.2 Rancangan pengujian kelas seimbang dan tidak seimbang

Pada rancangan pengujian ini berfungsi untuk mengetahui perbandingan antara tingkat hasil akurasi dari jumlah data latih pada kelas seimbang dengan jumlah data latih pada kelas tidak seimbang. Rancangan pengujian kelas seimbang dan tidak seimbang ditunjukkan pada Tabel 4.13.

Tabel 4.13 Rancangan Pengujian Kelas Seimbang Dan Tidak Seimbang

Jenis Data Latih	Jumlah Data Latih	Jumlah Data Uji	Rata-rata Akurasi (%)
Seimbang			
Tidak Seimbang			

4.6.3 Rancangan pengujian pengaruh jumlah data latih

Pada rancangan pengujian ini berfungsi untuk menguji pengaruh jumlah data latih yang berbeda terhadap jumlah data uji yang tetap pada hasil perbedaan tingkat akurasi. Sehingga dalam proses pengujian ini dilakukan dengan cara menambah jumlah data latih dengan jumlah data uji yang tetap. Rancangan pengujian pengaruh jumlah data latih ditunjukkan pada Tabel 4.14.

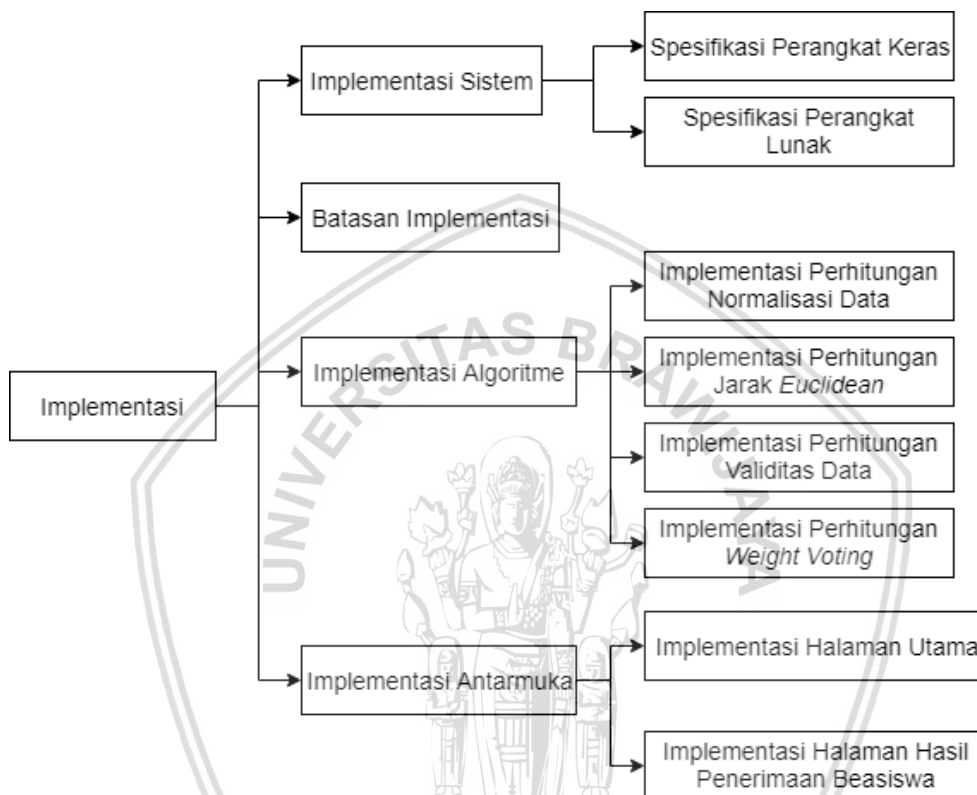
Tabel 4.14 Rancangan Pengujian Pengaruh Jumlah Data Latih

Jumlah Data Uji	Jumlah Data Latih	Nilai Akurasi (%)



BAB 5 IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi dari sistem yang dibangun, dimana pada implementasi sistem ini sendiri terdiri dari spesifikasi perangkat keras dan perangkat lunak, batasan implementasi, implementasi algoritme dan implementasi antarmuka. Berikut merupakan penjelasan dari implementasi pada Gambar 5.1.



Gambar 5.1 Diagram alir implementasi

5.1 Implementasi sistem

Implementasi sistem ini adalah merupakan kelanjutan dari analisis kebutuhan pada bab metodologi penelitian. Yang berfungsi untuk mengetahui kelayakan spesifikasi dari perangkat yang digunakan. Spesifikasi perangkat itu sendiri terdiri dari spesifikasi perangkat keras dan perangkat lunak.

5.1.1 Spesifikasi perangkat keras

Spesifikasi perangkat keras yang digunakan penelitian ini untuk mendukung proses dibangunnya sistem penentuan penerimaan beasiswa ditunjukkan pada Tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Processor	Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz (4 CPUs), ~2.7GHz
Memori	8192MB RAM
Kartu Grafis	NVIDIA Geforce 930MX
Harddisk	1000 GB

5.1.2 Spesifikasi perangkat lunak

Spesifikasi perangkat lunak yang digunakan penelitian ini untuk mendukung proses dibangunnya sistem penentuan penerimaan beasiswa ditunjukkan pada Tabel 5.2.

Tabel 5.2 Spesifikasi Perangkat Lunak

Nama	Spesifikasi
Sistem Operasi	Windows 10 64-bit
Bahasa Pemrograman	C#
Tool Pemrograman	Visual Studio 2015

5.2 Batasan implementasi

Pada implementasi dari sistem penentuan penerimaan beasiswa menggunakan metode *Modified K Nearest Neighbor* terdapat beberapa batasan diantaranya adalah sebagai berikut.

1. Sistem dibangun dengan menggunakan Bahasa pemrograman C#.
2. Metode yang digunakan dalam penelitian ini adalah metode *Modified K Nearest Neighbor*.
3. Masukan yang digunakan dalam sistem ini adalah data parameter pengajuan beasiswa yang dimasukkan oleh user.
4. Keluaran yang dihasilkan oleh sistem adalah status penerimaan beasiswa, perhitungan dengan metode *Modified K Nearest Neighbor* dan nilai akurasi dari perhitungan tersebut.

5.3 Implementasi algoritme

Untuk implementasi algoritme pada penelitian sistem penentuan penerimaan beasiswa dengan Algoritme *Modified K-Nearest Neighbor* ini menggunakan Bahasa pemrograman C#, pada sub bab ini akan ditampilkan implementasi algoritme perhitungan normalisasi data, perhitungan jarak *Euclidean*, perhitungan validitas data dan yang terakhir adalah perhitungan *weight voting*.

5.3.1 Implementasi perhitungan normalisasi data

Pada implementasi dari perhitungan normalisasi data dari penelitian ini adalah menggunakan normalisasi min-max, yang berfungsi untuk mempermudah proses perhitungan klasifikasi agar *range* data tidak berbeda jauh yaitu berada pada *range* [0,1], data yang akan dinormalisasi adalah data latih dan data uji sebelum nantinya akan diproses lebih lanjut dengan perhitungan klasifikasi *Modified K-Nearest Neighbor*. Implementasi dari perhitungan normalisasi data ditunjukkan pada *Source code* 5.1.

No	Source code
1	<code>private void btnNormalisasi_Click(object sender, EventArgs e)</code>
2	<code>{</code>
3	<code> lblProgNormalisasi.Visible = true;</code>
4	<code>if (gridDataLatihNormalisasi.Rows.Count > 0 </code>
5	<code>gridDataUjiNormalisasi.Rows.Count > 0)</code>
6	<code>{</code>
7	<code> MessageBox.Show("Normalisasi data telah</code>
8	<code>dilakukan.");</code>
9	<code>}</code>
10	<code>else</code>
11	<code>{</code>
12	<code> gridDataUji.Sort(gridDataUji.Columns[0],</code>
13	<code>ListSortDirection.Ascending);</code>
14	<code> gridDataLatih.Sort(gridDataLatih.Columns[0],</code>
15	<code>ListSortDirection.Ascending);</code>
16	<code></code>
17	<code> progNormalisasi.Maximum =</code>
18	<code>gridDataLatih.Rows.Count + gridDataUji.Rows.Count;</code>
19	<code> int c = 0;</code>
20	<code></code>
21	<code> gridDataLatihNormalisasi.Columns.Add("", "NO");</code>
22	<code> for (int i = 1; i <</code>
23	<code>gridDataLatih.Columns.Count-1; i++)</code>
24	<code> {</code>
25	<code> gridDataLatihNormalisasi.Columns.Add("",</code>
26	<code>"P" + ++c);</code>
27	<code> }</code>
28	<code> gridDataLatihNormalisasi.Columns.Add("",</code>
29	<code>"STATUS");</code>
30	<code> gridDataLatihNormalisasi.Rows.Add();</code>
31	<code></code>
32	<code> c = 0;</code>
33	<code></code>
34	<code> for (int i = 0; i < gridDataLatih.Rows.Count;</code>
35	<code>i++)</code>
36	<code> {</code>
37	<code> DataGridViewRow row =</code>
38	<code>(DataGridViewRow)gridDataLatihNormalisasi.Rows[0].Clone();</code>
39	<code> DataGridViewRow rowC =</code>
40	<code>gridDataLatih.Rows[i];</code>
41	<code></code>
42	<code> row.Cells[0].Value = ++c;</code>
43	<code> for (int j = 1; j <</code>
44	<code>gridDataLatih.Columns.Count; j++)</code>
45	<code> {</code>


```

46         if (j == 6)
47         {
48             row.Cells[j].Value =
49 rowC.Cells[j].Value.ToString();
50         }
51         else
52         {
53             double max =
54 gridDataLatih.Rows.Cast<DataGridViewRow>().Max(r =>
55 Convert.ToDouble(r.Cells[j].Value));
56             double min =
57 gridDataLatih.Rows.Cast<DataGridViewRow>().Min(r =>
58 Convert.ToDouble(r.Cells[j].Value));
59
60             row.Cells[j].Value =
61 Math.Round(((Convert.ToDouble(rowC.Cells[j].Value.ToString()) -
62 min) / (max - min)) * 1, 6);
63         }
64     }
65     gridDataLatihNormalisasi.Rows.Add(row);
66     progNormalisasi.Value++;
67     lblProgNormalisasi.Text =
68 progNormalisasi.Value + "/" + progNormalisasi.Maximum;
69 }
70
71 gridDataLatihNormalisasi.Rows.Remove(gridDataLatihNormalisasi.R
72 ows[0]);
73
74     c = 0;
75
76     gridDataUjiNormalisasi.Columns.Add("", "NO");
77     for (int i = 1; i < gridDataUji.Columns.Count -
78 1; i++)
79     {
80         gridDataUjiNormalisasi.Columns.Add("", "P"
81 + ++c);
82     }
83     gridDataUjiNormalisasi.Columns.Add("",
84 "STATUS");
85     gridDataUjiNormalisasi.Rows.Add();
86
87     c = 0;
88
89     for (int i = 0; i < gridDataUji.Rows.Count;
90 i++)
91     {
92         DataGridViewRow row =
93 (DataGridViewRow)gridDataUjiNormalisasi.Rows[0].Clone();
94         DataGridViewRow rowC = gridDataUji.Rows[i];
95
96         row.Cells[0].Value = ++c;
97         for (int j = 1; j <
98 gridDataUji.Columns.Count; j++)
99         {
100             if (j == 6)
101             {
102                 row.Cells[j].Value =
103 rowC.Cells[j].Value.ToString();
104             }

```

```

105         else
106         {
107             double max =
108 gridDataUji.Rows.Cast<DataGridViewRow>().Max(r =>
109 Convert.ToDouble(r.Cells[j].Value));
110             double min =
111 gridDataUji.Rows.Cast<DataGridViewRow>().Min(r =>
112 Convert.ToDouble(r.Cells[j].Value));
113
114             row.Cells[j].Value =
115 Math.Round(((Convert.ToDouble(rowC.Cells[j].Value.ToString()) -
116 min) / (max - min)) * 1, 6);
117         }
118     }
119     gridDataUjiNormalisasi.Rows.Add(row);
120     progNormalisasi.Value++;
121     lblProgNormalisasi.Text =
122 progNormalisasi.Value + "/" + progNormalisasi.Maximum;
123 }
124
125 gridDataUjiNormalisasi.Rows.Remove(gridDataUjiNormalisasi.Rows[
126 0]);
127     btnValiditasData.Enabled = true;
128 }
129 }
130

```

Source code 5.1 Normalisasi data

Penjelasan dari Source code 5.1 implementasi normalisasi data adalah sebagai berikut.

1. Baris 1 untuk membuat fungsi yang berguna dalam menjalankan sekumpulan kode program saat tombol Normalisasi ditekan.
2. Baris 3 untuk menampilkan status dari *ProgressBar*.
3. Baris 4-11 untuk menampilkan pesan normalisasi data telah dilakukan apabila gridDataLatihNormalisasi dan gridDataUjiNormalisasi tidak kosong dan telah terisi.
4. Baris 12-16 untuk mengurutkan data latih dan data uji dari terendah ke tertinggi.
5. Baris 17-20 untuk menentukan nilai maksimal dari *ProgressBar*.
6. Baris 21-31 untuk membuat header tabel data latih tiap kolom yang terdiri dari NO pada kolom pertama lalu P sejumlah 5 kemudian terakhir kolom status pada datagridview.
7. Baris 32-38 untuk membuat kloningan baris pada tabel dari header.
8. Baris 39-44 untuk membuat tabel yang dapat diisi dengan nilai.
9. Baris 46-50 untuk memberi nilai bertipe data string khusus pada kolom ke 6.
10. Baris 51-58 untuk mencari nilai tertinggi dan terendah dari setiap kolom data latih.
11. Baris 59-64 untuk memberikan nilai dengan memasukkan rumus min-max normalisasi.

12. Baris 65 untuk menambahkan nilai tiap baris dan melakukan perulangan hingga data latih terakhir.
13. Baris 66 untuk menambah nilai progress data yang telah dilakukan.
14. Baris 67-70 untuk menampilkan label tulisan progress dengan progress yang telah dilakukan dibandingkan dengan nilai maksimal dari progressbar.
15. Baris 71-72 untuk menghapus baris awal kloningan tabel.
16. Baris 73-85 untuk membuat header tabel data uji tiap kolom yang terdiri dari NO pada kolom pertama lalu P sejumlah 5 kemudian terakhir kolom status pada datagridview.
17. Baris 86-93 untuk membuat kloningan baris pada tabel dari header.
18. Baris 94-98 untuk membuat tabel yang dapat diisi dengan nilai.
19. Baris 99-106 untuk memberi nilai bertipe data string khusus pada kolom ke 6.
20. Baris 107-114 untuk mencari nilai tertinggi dan terendah dari setiap kolom data uji.
21. Baris 115-119 untuk memberikan nilai dengan memasukkan rumus min-max normalisasi.
22. Baris 120 untuk menambahkan nilai tiap baris dan melakukan perulangan hingga data uji terakhir.
23. Baris 121 untuk menambah nilai progress data yang telah dilakukan.
24. Baris 122-125 untuk menampilkan label tulisan progress dengan progress yang telah dilakukan dibandingkan dengan nilai maksimal dari progressbar.
25. Baris 126-127 untuk menghapus baris awal kloningan tabel.
26. Baris 128 untuk mengaktifkan tombol *tab control* validitas untuk melanjutkan proses perhitungan.

5.3.2 Implementasi algoritme perhitungan Euclidean

Pada implementasi perhitungan *Euclidean* dari penelitian ini berfungsi untuk mengetahui jarak *Euclidean* antar data latih yang nantinya akan digunakan dalam melakukan perhitungan validitas dan jarak *Euclidean* antara data latih dan data uji yang nantinya akan digunakan dalam perhitungan *weight voting*. Implementasi dari algoritme perhitungan normalisasi data ditunjukkan pada *Source code* 5.2.

No	Source code
1	<code>private void btnValiditasData_Click(object sender, EventArgs e)</code>
2	<code>{</code>
3	<code> lblProgValiditas.Visible = true;</code>
4	<code></code>
5	<code>if (gridValiditasData.Rows.Count > 0 </code>
6	<code>gridEuclideanDataLatih.Rows.Count > 0)</code>
7	<code>{</code>
8	<code> MessageBox.Show("Validitas data telah</code>
9	<code>dilakukan.");</code>
10	<code>}</code>
11	<code>else</code>
12	<code>{</code>

```

13         gridDataUji.Sort(gridDataUji.Columns[0],
14 ListSortDirection.Ascending);
15         gridDataLatih.Sort(gridDataLatih.Columns[0],
16 ListSortDirection.Ascending);
17
18 gridDataLatihNormalisasi.Sort(gridDataLatihNormalisasi.Columns[
19 0], ListSortDirection.Ascending);
20
21 gridDataUjiNormalisasi.Sort(gridDataUjiNormalisasi.Columns[0],
22 ListSortDirection.Ascending);
23
24         int c = 0;
25         progValiditas.Maximum =
26 gridDataLatih.Rows.Count + gridDataLatih.Rows.Count;
27
28         gridEuclideanDataLatih.Columns.Add("", "DATA");
29         for (int i = 0; i < gridDataLatih.Rows.Count;
30 i++)
31         {
32             gridEuclideanDataLatih.Columns.Add("", "D"
33 + ++c);
34         }
35         gridEuclideanDataLatih.Columns.Add("",
36 "STATUS");
37         gridEuclideanDataLatih.Columns.Add("", "no");
38
39 gridEuclideanDataLatih.Columns[gridEuclideanDataLatih.Columns.C
40 ount-1].Visible = false;
41         gridEuclideanDataLatih.Rows.Add();
42
43         c = 0;
44         for (int i = 0; i < gridDataLatih.Rows.Count;
45 i++)
46         {
47             DataGridViewRow rowE =
48 (DataGridViewRow)gridEuclideanDataLatih.Rows[0].Clone();
49             DataGridViewRow row =
50 gridDataLatihNormalisasi.Rows[i];
51             int cint = 0;
52
53             rowE.Cells[0].Value = "D"+ ++c;
54             for (int j = 0; j <
55 gridDataLatih.Rows.Count; j++)
56             {
57                 DataGridViewRow rowC =
58 gridDataLatihNormalisasi.Rows[j];
59                 double finalCount = 0;
60
61                 for (int m = 1; m <= 5; m++)
62                 {
63                     double times =
64 Convert.ToDouble(row.Cells[m].Value.ToString()) -
65 Convert.ToDouble(rowC.Cells[m].Value.ToString());
66                     times = Math.Pow(times, 2);
67
68                     finalCount += times;
69                 }
70                 finalCount = Math.Round(finalCount, 6);
71

```

```

72         rowE.Cells[++cint].Value =
73         Math.Round(Math.Sqrt(finalCount), 6);
74     }
75     rowE.Cells[gridDataLatih.Rows.Count +
76     1].Value = row.Cells[6].Value.ToString();
77     rowE.Cells[gridDataLatih.Rows.Count +
78     2].Value = c;
79     gridEuclideanDataLatih.Rows.Add(rowE);
80     progValiditas.Value++;
81     lblProgValiditas.Text = progValiditas.Value
82     + "/" + progValiditas.Maximum;
83 }
84 gridEuclideanDataLatih.Rows.Remove(gridEuclideanDataLatih.Rows[
85 0]);
86

```

Source code 5.2 Jarak euclidean

Penjelasan dari Source code 5.2 implementasi normalisasi data adalah sebagai berikut

1. Baris 1 untuk membuat fungsi yang berguna dalam menjalankan sekumpulan kode program saat tombol validitas ditekan.
2. Baris 2-4 untuk menampilkan status dari *ProgressBar*.
3. Baris 4-10 untuk menampilkan pesan validitas data telah dilakukan apabila *gridDataLatihNormalisasi* dan *gridDataUjiNormalisasi* tidak kosong dan telah terisi.
4. Baris 11-18 untuk mengurutkan data latih dan data uji dari terendah ke tertinggi.
5. Baris 19-24 untuk mengurutkan data latih dan data uji yang telah dinormalisasi dari terendah ke tertinggi.
6. Baris 25-28 untuk menentukan nilai maksimal dari *ProgressBar*.
7. Baris 29-39 untuk membuat header tabel.
8. Baris 40-42 untuk membuat kolom terakhir "no" tidak terlihat
9. Baris 42 untuk memasukkan baris header kedalam *datagrid*
10. Baris 43-50 untuk membuat baris kloningan dari header.
11. Baris 51-52 untuk memanggil *datagrid* normalisasi data latih.
12. Baris 54-55 untuk melakukan perulangan.
13. Baris 56-62 untuk memanggil data latih normalisasi.
14. Baris 63-64 untuk parameter p1 sampai p5.
15. Baris 65-71 untuk menghitung jarak antar data latih.
16. Baris 72 untuk membuat hasil perhitungan jarak 6 angka dibelakang koma
17. Baris 73-74 untuk menghitung akar pangkat 2 dari hasil perhitungan jarak data latih.
18. Baris 75-79 untuk mendapatkan nilai dari status.
19. Baris 80 untuk menambahkan data *datagrid* jarak *Euclidean* data latih.
27. Baris 81 untuk menambah nilai progress data yang telah dilakukan.

20. Baris 82-84 untuk menampilkan label tulisan progress dengan progress yang telah dilakukan dibandingkan dengan nilai maksimal dari progressbar.
21. Baris 85-86 untuk menghapus baris awal kloningan tabel.

5.3.3 Implementasi algoritme perhitungan validitas

Pada implementasi perhitungan validitas dari penelitian ini dilakukan dengan melakukan perbandingan antara status penerimaan pada data latih sesuai dengan data sebanyak nilai k yang digunakan, apabila kelas yang dibandingkan berada pada status yang sama maka akan diberi nilai 1 dan sebaliknya apabila berada pada status yang berbeda maka akan diberi nilai 0. Implementasi dari algoritme perhitungan normalisasi data ditunjukkan pada *Source code* 5.3.

No	Source code
1	<code>c = 0;</code>
2	
3	<code>gridValiditasData.Columns.Add("", "NO");</code>
4	<code>for (int i = 0; i < k; i++)</code>
5	<code>{</code>
6	<code> gridValiditasData.Columns.Add("", "K" + ++c);</code>
7	<code>}</code>
8	<code>gridValiditasData.Columns.Add("", "SUM");</code>
9	<code>gridValiditasData.Columns.Add("", "Validitas");</code>
10	<code>gridValiditasData.Rows.Add();</code>
11	
12	<code>c = 0;</code>
13	<code>for (int i = 0; i < gridEuclideanDataLatih.Rows.Count;</code>
14	<code>i++)</code>
15	<code>{</code>
16	<code> DataGridViewRow row =</code>
17	<code>(DataGridViewRow)gridValiditasData.Rows[0].Clone();</code>
18	<code> DataGridViewRow rowN =</code>
19	<code>gridEuclideanDataLatih.Rows[i];</code>
20	<code> int sum = 0;</code>
21	
22	<code> row.Cells[0].Value = ++c;</code>
23	<code> for (int j = 1; j <= k; j++) {</code>
24	<code> var cal =</code>
25	<code>gridEuclideanDataLatih.Rows.Cast<DataGridViewRow>().OrderBy(r =></code>
26	<code>Convert.ToDouble(r.Cells[c].Value)).Skip(j).FirstOrDefault();</code>
27	<code> String status =</code>
28	<code>cal.Cells[gridEuclideanDataLatih.Rows.Count+1].Value.ToString();</code>
29	
30	<code> if</code>
31	<code>(rowN.Cells[gridEuclideanDataLatih.Rows.Count+1].Value.Equals(status))</code>
32	<code> {</code>
33	<code> sum++;</code>
34	<code> row.Cells[j].Value = 1;</code>
35	<code> }</code>
36	<code> else</code>
37	<code> {</code>
38	<code> row.Cells[j].Value = 0;</code>
39	<code> }</code>
40	
41	<code> row.Cells[j + 1].Value = sum;</code>
42	


```

43         row.Cells[j + 2].Value =
44         Math.Round((double) sum / k, 3);
45     }
46
47     gridValiditasData.Rows.Add(row);
48     progValiditas.Value++;
49     lblProgValiditas.Text = progValiditas.Value + "/"
50 + progValiditas.Maximum;
51     }
52
53     gridValiditasData.Rows.Remove(gridValiditasData.Rows[0]);
54     btnWeightVoting.Enabled = true;
55     }

```

Source code 5.3 Validitas

Penjelasan dari Source code 5.3 implementasi normalisasi data adalah sebagai berikut

1. Baris 1-11 untuk membuat baris header dari *datagrid* validitas.
2. Baris 12-17 untuk membuat kloningan baris dari header pada tabel *datagrid*.
3. Baris 18-19 untuk memanggil jarak *Euclidean* data latih.
4. Baris 20-29 untuk mengurutkan jarak *Euclidean* data latih dari yang paling terendah sebanyak hingga maksimal dari nilai k.
5. Baris 30-40 untuk mengambil data string status apabila sama akan bernilai 1 dan jika tidak bernilai 0.
6. Baris 41- 45 untuk menjumlahkan total nilai dan membagi dengan nilai k.
7. Baris 46 untuk memasukkan data ke *datagrid* validitas.
8. Baris 47 untuk menambah nilai progress data yang telah dilakukan.
9. Baris 48-51 untuk menampilkan label tulisan progress dengan progress yang telah dilakukan dibandingkan dengan nilai maksimal dari progressbar.
10. Baris 52 untuk menghapus baris awal kloningan tabel.
11. Baris 53 untuk mengaktifkan tombol *tab control weight voting* untuk melanjutkan proses perhitungan.

5.3.4 Implementasi algoritme perhitungan weight voting

Pada implementasi perhitungan *weight voting* dari penelitian ini dilakukan berdasarkan bobot hasil perhitungan jarak *Euclidean* antara data latih dan data uji dengan hasil perhitungan validitas, implementasi ini berfungsi untuk menentukan status penerimaan beasiswa mahasiswa apakah diterima atau ditolak. Berikut adalah merupakan implementasi dari algoritme perhitungan normalisasi data pada Source code 5.4.

No	Source code
1	private void btnWeightVoting_Click(object sender, EventArgs e)
2	{
3	lblProgWeightVoting.Visible = true;
4	}


```

5         if (gridJarakDataLatihDataUji.Rows.Count > 0 ||
6         gridWeightVoting.Rows.Count > 0)
7         {
8             MessageBox.Show("Weight voting data telah dilakukan.");
9         }
10        else
11        {
12            gridDataUji.Sort(gridDataUji.Columns[0], ListSortDirection.Ascending);
13            gridDataLatih.Sort(gridDataLatih.Columns[0]ListSortDirection.Ascending);
14            gridDataLatihNormalisasi.Sort(gridDataLatihNormalisasi.Columns[0],
15            ListSortDirection.Ascending);
16            gridDataUjiNormalisasi.Sort(gridDataUjiNormalisasi.Columns[0],
17            ListSortDirection.Ascending);
18            gridEuclideanDataLatih.Sort(gridEuclideanDataLatih.Columns[gridEuclidean
19            ataLatih.Columns.Count-1], ListSortDirection.Ascending);
20            gridValiditasData.Sort(gridValiditasData.Columns[0],ListSortDirection.Asc
21            ending);
22            int c = 0;
23            progWeightVoting.Maximum = gridDataLatih.Rows.Count +
24            gridDataLatih.Rows.Count;
25
26            gridJarakDataLatihDataUji.Columns.Add("", "NO");
27            for (int i = 0; i < gridDataUji.Rows.Count; i++)
28            {
29                gridJarakDataLatihDataUji.Columns.Add("", "" + ++c);
30            }
31            gridJarakDataLatihDataUji.Rows.Add();
32            c = 0;
33            for (int i = 0; i < gridDataLatih.Rows.Count; i++)
34            {
35                DataGridViewRow rowE =
36                (DataGridViewRow)gridJarakDataLatihDataUji.Rows[0].Clone();
37                DataGridViewRow row = gridDataLatihNormalisasi.Rows[i];
38
39                rowE.Cells[0].Value = ++c;
40                for (int j = 0; j < gridDataUji.Rows.Count; j++)
41                {
42                    DataGridViewRow rowC = gridDataUjiNormalisasi.Rows[j];
43                    double finalCount = 0;
44
45                    for (int m = 1; m <= 5; m++)
46                    {
47                        double times = Convert.ToDouble(row.Cells[m].Value.ToString()) -
48                        Convert.ToDouble(rowC.Cells[m].Value.ToString());
49                        times = Math.Pow(times, 2);
50                        finalCount += times;
51                    }
52                    finalCount = Math.Round(finalCount,6);
53                    rowE.Cells[j+1].Value =
54                    Math.Round(Math.Sqrt(finalCount), 6);
55                }
56                gridJarakDataLatihDataUji.Rows.Add(rowE);
57                progWeightVoting.Value++;
58                lblProgWeightVoting.Text = progWeightVoting.Value + "/" +
59                progWeightVoting.Maximum;}
60            gridJarakDataLatihDataUji.Rows.Remove(gridJarakDataLatihDataUji.Rows[0]);
61            c = 0;
62            gridWeightVoting.Columns.Add("", "NO");
63            for (int i = 0; i < gridDataUji.Rows.Count; i++)

```

```

64         {
65             gridWeightVoting.Columns.Add("", ""+ ++c);
66         }
67         gridWeightVoting.Columns.Add("", "STATUS");
68         gridWeightVoting.Rows.Add();
69
70         c = 0;
71         for (int i = 0; i < gridDataLatih.Rows.Count; i++)
72         {
73             DataGridViewRow rowE =
74 (DataGridViewRow)gridWeightVoting.Rows[0].Clone();
75             DataGridViewRow row = gridValiditasData.Rows[i];
76             DataGridViewRow rowC = gridJarakDataLatihDataUji.Rows[i];
77
78             rowE.Cells[0].Value = ++c;
79             int cint = 0;
80
81             for (int m = 1; m <= gridDataUji.Rows.Count; m++)
82             {
83                 double times = 1 / (Convert.ToDouble(rowC.Cells[m].Value.ToString()) +
84 alpha);
85                 times *= Convert.ToDouble(row.Cells[gridValiditasData.Columns.Count-
86 1].Value.ToString());
87
88                 rowE.Cells[m].Value = Math.Round(times, 6);
89                 cint = m;
90             }
91             rowE.Cells[cint + 1].Value =
92 gridEuclideanDataLatih.Rows[i].Cells[gridDataLatih.Rows.Count+1].Value;
93             gridWeightVoting.Rows.Add(rowE);
94             progWeightVoting.Value++;
95             lblProgWeightVoting.Text = progWeightVoting.Value +
96 "/" + progWeightVoting.Maximum;}
97             gridWeightVoting.Rows.Remove(gridWeightVoting.Rows[0]);
98             btnHasilAkhir.Enabled = true; } }

```

Source code 5.4 Weight voting

Penjelasan dari Source code 5.4 implementasi normalisasi data adalah sebagai berikut

1. Baris 1 untuk membuat fungsi yang berguna dalam menjalankan sekumpulan kode program saat tombol *weight voting* ditekan.
2. Baris 2-3 untuk menampilkan status dari *ProgressBar*.
3. Baris 4-8 untuk menampilkan pesan *weight voting* data telah dilakukan apabila *gridDataLatihNormalisasi* dan *gridDataUjiNormalisasi* tidak kosong dan telah terisi.
4. Baris 9-20 untuk mengurutkan data.
5. Baris 21-23 untuk menentukan nilai maksimal dari *ProgressBar*.
6. Baris 24-31 untuk membuat header pada *data grid* jarak data uji dan data latih.
7. Baris 32-35 untuk membuat kloningan baris pada tabel.
8. Baris 36-53 untuk melakukan perhitungan jarak *Euclidean* antara data uji dan data latih.

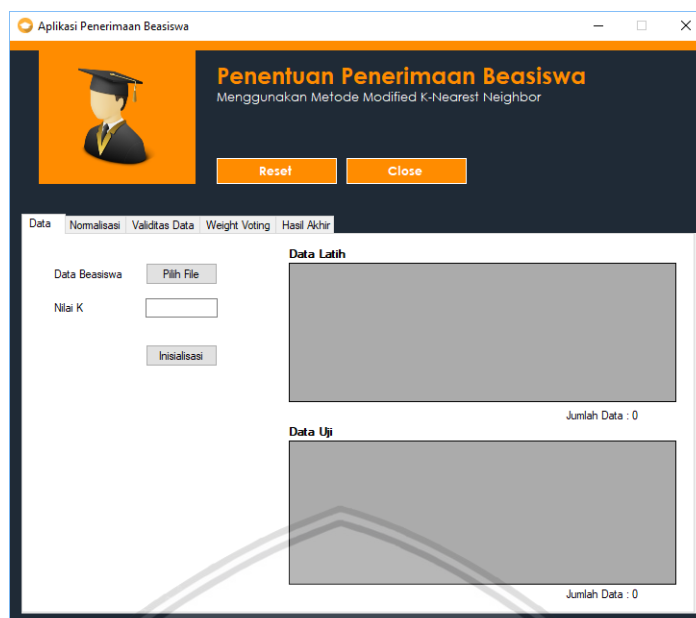
9. Baris 55 untuk memasukkan ke *datagrid* jarak data uji dan data latih.
10. Baris 56 untuk menambah nilai progress data yang telah dilakukan.
12. Baris 57-58 untuk untuk menampilkan label tulisan progress dengan progress yang telah dilakukan dibandingkan dengan nilai maksimal dari progressbar.
11. Baris 59 untuk menghapus baris kloningan tabel.
12. Baris 60-71 untuk membuat header *datagrid weight voting*.
13. Baris 72-73 untuk membuat kloningan baris pada tabel.
14. Baris 74-75 untuk inialisasi data.
15. Baris 76-89 untuk melakukan perhitungan *weight voting*.
16. Baris 90-92 untuk memasukkan data pada *datagrid weight voting*.
17. Baris 93 untuk menambah nilai progress data yang telah dilakukan.
18. Baris 94-95 untuk untuk untuk menampilkan label tulisan progress dengan progress yang telah dilakukan dibandingkan dengan nilai maksimal dari progressbar.
19. Baris 96 untuk menghapus baris kloningan.
20. Baris 97 untuk mengaktifkan tombol *tab control* hasil akhir untuk melanjutkan proses klasifikasi mengetahui hasil dari klasifikasi.

5.4 Implementasi antarmuka

Untuk antarmuka dari sistem penentuan penerimaan beasiswa dengan metode *Modified K-Nearest Neighbor* pada penelitian ini adalah sebagai sarana antara pengguna dan sistem agar dapat berinteraksi sehingga pengguna dapat menggunakan sistem dengan baik dan mendapat informasi tentang status penerimaan beasiswa. Pada sub bab ini berisi implementasi antarmuka halaman utama dan implementasi antarmuka halaman hasil penerimaan beasiswa.

5.4.1 Implementasi antarmuka halaman utama

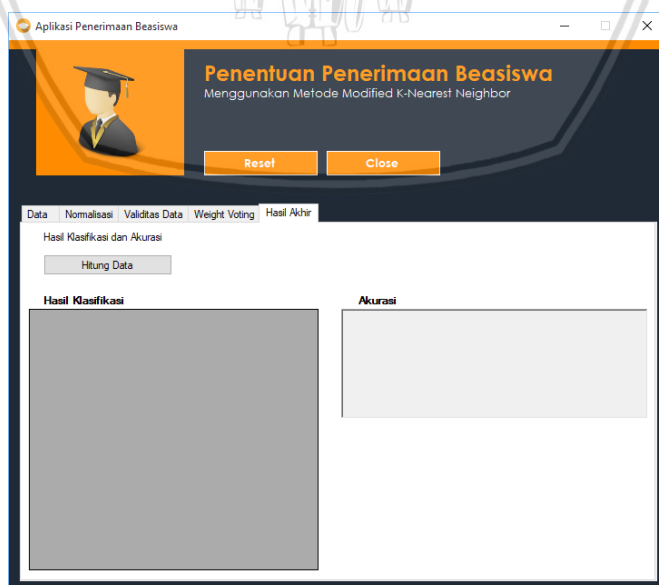
Implementasi antarmuka halaman utama pada penelitian ini, terdapat nama aplikasi pada bagian atas yaitu penerimaan beasiswa. Kemudian dibawahnya terdapat tombol *reset* yang berfungsi untuk mengulang aplikasi dari awal yaitu dari *tab control* data dan tombol *close* untuk menutup aplikasi. Terdapat beberapa *tab control* yang bisa dijalankan, pada *tab control* data terdapat tombol untuk memilih file Data Uji beserta Data Latih dengan format .xls, selanjutnya terdapat *textbox* yang berfungsi untuk memasukkan nilai k. *Button* inialisasi berguna untuk memasukkan data ke sistem dan selanjutnya di tampilkan pada *datagridview* berupa data yang telah masuk ke dalam sistem dalam kolom Data Uji dan kolom Data Latih, Tampilan implementasi antarmuka halaman utama ditunjukkan pada gambar 5.2.



Gambar 5.2 Implementasi Antarmuka Halaman Utama

5.4.2 Implementasi antarmuka halaman hasil penerimaan beasiswa

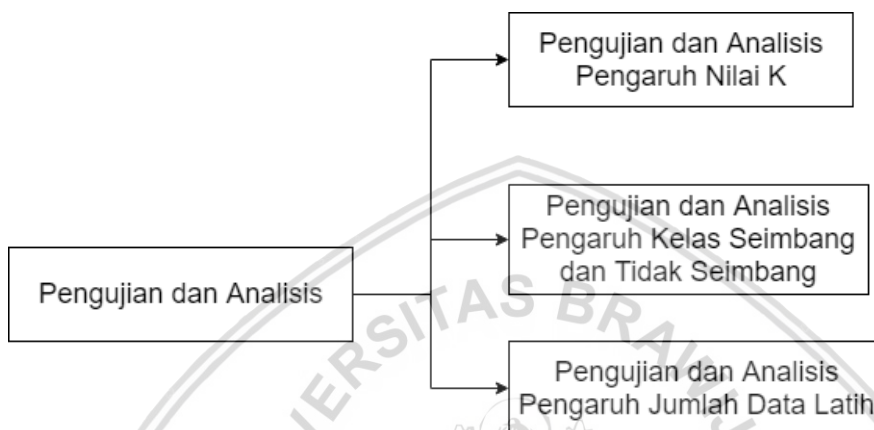
Implementasi antarmuka halaman hasil penerimaan beasiswa pada penelitian ini, terdapat tombol untuk melakukan perhitungan hasil dan akurasi dengan syarat perhitungan pada bagian sebelumnya telah dilalui. Terdapat dua bagian yang pertama *datagridview* hasil status klasifikasi sistem dan data uji, yang kedua adalah *richtextbox* hasil akurasi dari sistem yang dilakukan dengan membandingkan hasil klasifikasi sistem dengan data uji yang ada. Tampilan implementasi antarmuka halaman hasil penerimaan beasiswa ditunjukkan pada gambar 5.3.



Gambar 5.3 Implementasi antarmuka halaman hasil penerimaan beasiswa

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan dijelaskan mengenai pengujian dan analisis dari sistem Penentuan Penerimaan Beasiswa menggunakan metode *Modified K-Nearest Neighbor*. Terdapat tiga pengujian yang akan dilakukan, diantaranya adalah meliputi pengujian pengaruh nilai k, pengujian pengaruh kelas seimbang dan tidak seimbang, pengujian pengaruh jumlah data latih terhadap akurasi sistem yang dihasilkan. Diagram alir dari pengujian dan analisis ditunjukkan pada Gambar 6.1.



Gambar 6.1 Diagram alir pengujian dan analisis

6.1 Pengujian pengaruh nilai k

Pada pengujian ini akan dilakukan dengan cara memasukkan nilai k yang berbeda ke sistem untuk mengetahui pengaruh nilai k terhadap nilai hasil akurasi, nilai k yang digunakan adalah $k=1$ sampai dengan $k=10$. Untuk komposisi data yang digunakan dalam pengujian ini adalah 5 kali percobaan dari data latih sebanyak 20, 30 dan 40 data dengan data uji tetap sebanyak 185 data.

6.1.1 Skenario pengujian pengaruh nilai k

Tujuan dari pengujian ini adalah untuk dapat mengetahui seberapa besar pengaruh nilai k terhadap nilai hasil akurasi, untuk nilai k yang akan digunakan dalam penelitian ini adalah sebesar $k=1$ sampai dengan $k=10$. Pengujian ini dibagi menjadi tiga uji coba, yaitu adalah uji coba pada data latih 20, 30 dan 40 data dengan jumlah data uji yang tetap sebanyak 185 data. Total keseluruhan jumlah dataset yang digunakan dalam penelitian ini adalah sebanyak 370 data. Untuk setiap uji coba dilakukan sebanyak masing-masing 5 kali dengan komposisi data yang berbeda, kemudian diambil nilai rata-rata hasil akurasi dari keseluruhan uji coba yang telah dilakukan.

6.1.2 Analisis hasil pengujian pengaruh nilai k

Berdasarkan pengujian dengan tiga uji coba yang berbeda sebanyak 5 kali setiap dari nilai k dari 1 sampai dengan 10, didapatkan hasil akurasi yang ditunjukkan pada Tabel 6.1. Pada tabel tersebut dapat diketahui bahwa terlihat perbedaan akurasi yang dihasilkan dengan menggunakan nilai k yang berbeda

pada tiga uji coba yang berbeda tingkat data ujinya. Hasil akurasi pengujian pengaruh nilai k pada data latih 20, data latih 30 dan data latih 40 ditunjukkan pada Tabel 6.1.

Tabel 6.1 Hasil pengujian pengaruh nilai k

Nilai K	Uji Coba	Akurasi (%)		
		Data Latih 20	Data Latih 30	Data Latih 40
1	1	55	63	62
	2	85	69	66
	3	81	66	67
	4	61	69	67
	5	51	74	57
2	1	50	64	86
	2	62	62	85
	3	81	68	87
	4	83	66	87
	5	68	74	85
3	1	85	74	85
	2	85	65	88
	3	69	77	87
	4	72	66	87
	5	57	84	85
4	1	85	82	83
	2	85	78	86
	3	78	84	87
	4	79	68	85
	5	74	83	86
5	1	86	88	85
	2	85	86	88
	3	86	86	87
	4	88	85	88
	5	83	88	87
6	1	86	88	72

	2	85	81	87
	3	79	82	82
	4	84	86	87
	5	75	82	88
7	1	83	86	82
	2	85	80	83
	3	56	84	84
	4	59	83	87
	5	51	79	87
8	1	84	88	65
	2	85	75	70
	3	70	74	82
	4	77	81	87
	5	59	80	86
9	1	86	88	55
	2	85	71	63
	3	74	72	82
	4	81	84	87
	5	72	81	87
10	1	86	88	65
	2	85	66	68
	3	80	68	81
	4	84	84	87
	5	74	86	87

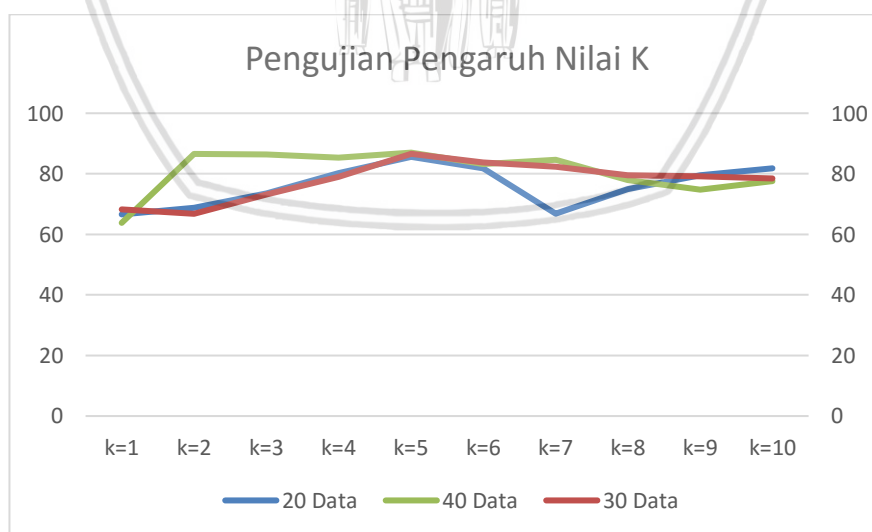
Setelah didapatkan nilai hasil akurasi dari 5 kali uji coba pada data latih 20, data latih 30 dan data latih 40 dengan nilai k yang berbeda, rata-rata akurasi pengujian pengaruh nilai k pada Tabel 6.2.

Tabel 6.2 Rata-rata hasil pengujian pengaruh nilai k

Nilai K	Rata-Rata Akurasi (%)		
	Data Latih 20	Data Latih 30	Data Latih 40
1	66.6	68.2	63.8
2	68.8	66.8	86

3	73.6	73.2	86.4
4	80.2	79	85.4
5	85.6	86.6	87
6	81.8	83.8	83.2
7	66.8	82.4	84.6
8	75	79.6	78
9	79.6	79.2	74.8
10	81.8	78.4	77.6

Berdasarkan uji coba nilai hasil rata-rata akurasi pada Tabel 6.2, didapatkan hasil rata-rata akurasi tertinggi pada nilai $k=5$. Disimpulkan bahwa nilai k yang digunakan memiliki pengaruh terhadap hasil dari akurasi sistem. *Error* terjadi pada semakin tinggi nilai k dari nilai k terbaik menyebabkan menurunnya hasil nilai akurasi, hal tersebut dikarenakan semakin besar nilai k maka semakin banyak pula tetangga terdekat yang digunakan sebagai pertimbangan pengambilan keputusan dalam proses klasifikasi sehingga dapat menyebabkan *noise*. Dan juga, apabila semakin rendah nilai k dari nilai k terbaik maka semakin akan sedikit tetangga terdekat yang digunakan dalam proses klasifikasi sehingga berdampak pada nilai akurasi yang dihasilkan lebih rendah. Nilai hasil akurasi tertinggi terletak pada nilai $k=5$. Berdasarkan pada hasil dari tiap kenaikan nilai mengalami peningkatan hingga nilai $k=5$ namun kemudian setelah itu nilai akurasi k menurun seiring peningkatan nilai k dari nilai k terbaik. Maka nilai k terbaik yang digunakan adalah yaitu $k=5$ yang digunakan untuk pengujian selanjutnya. Grafik hasil pengujian pengaruh nilai k ditunjukkan pada Gambar 6.2.



Gambar 6.2 Grafik hasil pengujian pengaruh nilai k

6.2 Pengujian kelas seimbang dan tidak seimbang

Pada pengujian kelas seimbang dan tidak seimbang ini digunakan untuk mengetahui pengaruh dari persebaran data yang seimbang dan tidak seimbang terhadap nilai dari hasil akurasi yang dihasilkan sistem. Untuk nilai k yang digunakan dalam pengujian ini bernilai $k=5$, dikarenakan berdasar dari pengujian sebelumnya memiliki nilai akurasi yang paling tinggi. Data yang akan digunakan di pengujian ini adalah data latih sebanyak 20, 30 dan 40 data.

6.2.1 Skenario pengujian kelas seimbang dan tidak seimbang

Pada pengujian kelas seimbang dan tidak seimbang ini bertujuan untuk mengetahui hasil akurasi persebaran data latih yang seimbang dan tidak seimbang sedangkan data uji tetap. Pengujian ini dilakukan dengan data latih sebanyak 20, 30 dan 40 dengan tiap uji coba dilakukan sebanyak 5 kali dengan data uji yang sama tetap yaitu sebanyak 185 data. Nilai k yang digunakan adalah $k=5$ berdasarkan nilai k yang memiliki tingkat akurasi tertinggi pada pengujian sebelumnya. Untuk pengujian kelas seimbang dilakukan dengan komposisi data latih dengan jumlah yang sama pada setiap kelasnya. Seperti contoh untuk data latih 40 maka data tersebut terdiri dari 20 kelas status diterima dan 20 kelas status ditolak. Sebaliknya, untuk pengujian kelas tidak seimbang dilakukan dengan komposisi data yang berbeda. Data uji yang digunakan dalam pengujian ini sebanyak 185 data.

6.2.2 Analisis hasil pengujian kelas seimbang dan tidak seimbang

Setelah pengujian dilakukan sebanyak 5 kali tiap uji coba pada persebaran data seimbang dan tidak seimbang dengan data latih sebanyak 20, 30 dan 40 dan data uji tetap 185 data. Perbedaan akurasi yang dihasilkan menunjukkan bahwa sebaran data dapat mempengaruhi tingkat hasil akurasi. Berikut merupakan hasil pengujian kelas seimbang dan tidak seimbang pada Tabel 6.3.

Tabel 6.3 Hasil pengujian kelas seimbang dan tidak seimbang

Jenis Data	Data Latih	Data Uji	Uji Coba	Akurasi(%)
Seimbang	20	185	1	86
			2	85
			3	86
			4	88
			5	82
	30	185	1	88
			2	85
			3	86
			4	85

	40	185	5	88
			1	85
			2	88
			3	87
			4	87
			5	86
Tidak Seimbang	20	185	1	69
			2	80
			3	88
			4	86
			5	87
	30	185	1	85
			2	70
			3	87
			4	86
			5	61
	40	185	1	85
			2	87
			3	79
			4	87
			5	73

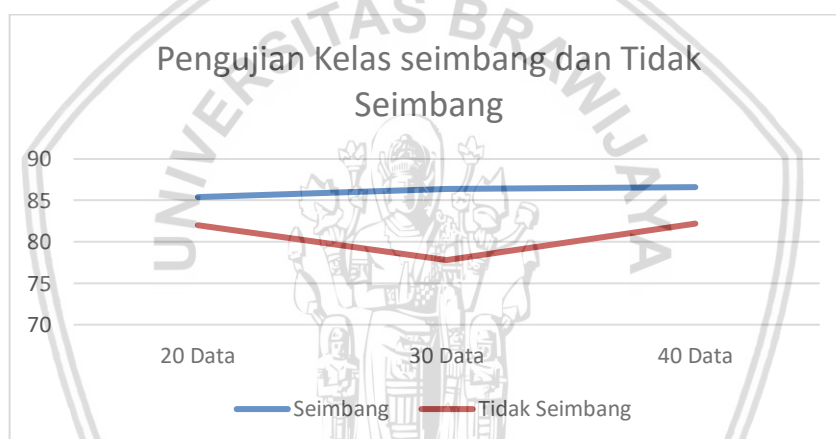
Hasil nilai akurasi dari 5 kali uji coba pengujian pengaruh sebaran data dari jumlah data latih yang berbeda dan data uji tetap pada tabel 6.3. Hasil rata-rata pengujian kelas seimbang dan kelas tidak seimbang pada tabel 6.4.

Tabel 6.4 Rata-rata hasil pengujian kelas seimbang dan tidak seimbang

Jenis Data	Data Latih	Data Uji	Rata-rata Akurasi(%)
Seimbang	20	185	85.4
	30		86.4
	40		86.6
Tidak Seimbang	20	185	82
	30		77.8

	40		82.2
--	----	--	------

Berdasarkan hasil uji coba yang ditunjukkan pada Tabel 6.4, diketahui hasil rata-rata akurasi tertinggi sebesar 86.6% pada data latih seimbang sebanyak 40 data latih dan hasil rata-rata akurasi terendah sebesar 77.8% pada data latih tidak seimbang sebanyak 30 data latih. Dapat disimpulkan bahwa data latih seimbang memiliki akurasi yang cenderung lebih tinggi dan stabil apabila dibandingkan dengan data latih tidak seimbang. Akurasi yang dihasilkan oleh data latih seimbang mengalami peningkatan seiring dengan meningkatnya jumlah data latih, sedangkan pada data latih tidak seimbang hasil akurasi yang dihasilkan mengalami peningkatan dan penurunan pada setiap kenaikan jumlah data latih. Hal tersebut disebabkan oleh adanya kelas yang mendominasi pada data latih tidak seimbang, sehingga kemungkinan untuk terjadinya *noise* dalam proses klasifikasi dan hasil yang didapatkan menjadi tidak akurat. Grafik hasil pengujian kelas seimbang dan tidak seimbang pada Gambar 6.3.



Gambar 6.3 Grafik hasil pengujian kelas seimbang dan tidak seimbang

6.3 Pengujian pengaruh jumlah data latih

Pada pengujian pengaruh jumlah data latih ini dilakukan dengan menggunakan data latih sebanyak 20, 30 dan 40. Yang bertujuan untuk mengetahui pengaruh dari jumlah data latih yang digunakan sedangkan untuk data uji berjumlah tetap yaitu sebanyak 185 data. Nilai k yang digunakan dalam pengujian ini adalah $k=5$.

6.3.1 Skenario pengujian pengaruh jumlah data latih

Pengujian ini bertujuan untuk mengetahui seberapa besar pengaruh dari jumlah data latih. Dilakukan dengan data latih yang berjumlah berbeda, yaitu sebanyak 20, 30 dan 40 data latih. Dengan menggunakan data uji yang tetap yaitu sebanyak 185 data dan nilai k yang bernilai $k=5$ dikarenakan berdasarkan pengujian sebelumnya nilai k tersebut merupakan nilai akurasi tertinggi. Pada pengujian ini akan dilakukan sebanyak 5 kali percobaan, yang nantinya hasilnya akan diambil rata-ratanya untuk dianalisis lebih lanjut.

6.3.2 Analisis hasil pengujian pengaruh jumlah data latih

Setelah dilakukan pengujian sebanyak 5 kali untuk setiap data pada data latih sebanyak 20, 30 dan 40. Hasil akurasi yang didapatkan menunjukkan bahwa jumlah data latih yang berbeda mempengaruhi nilai hasil akurasi. Hasil pengujian pengaruh jumlah data latih pada Tabel 6.5.

Tabel 6.5 Hasil pengujian pengaruh jumlah data latih

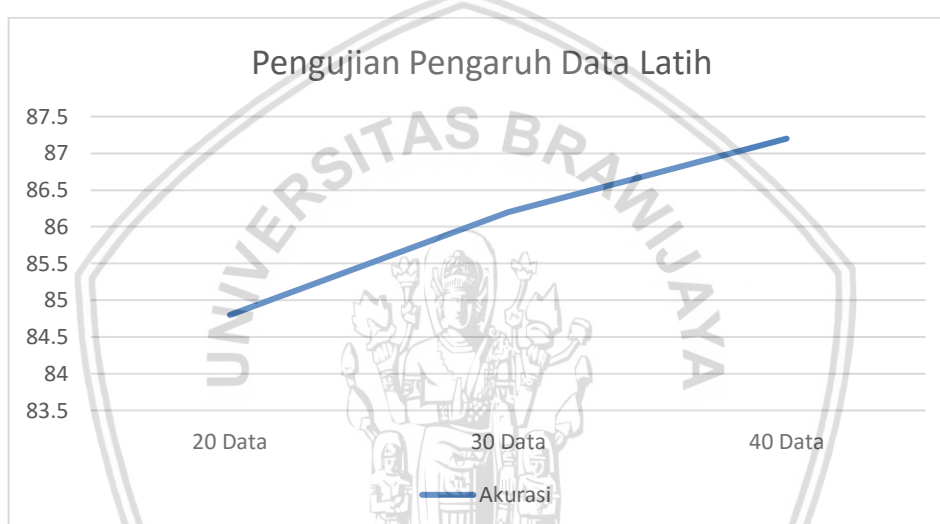
Data Latih	Data Uji	Uji Coba	Akurasi(%)
20	120	1	88
		2	84
		3	85
		4	86
		5	81
30	120	1	87
		2	84
		3	86
		4	86
		5	88
40	120	1	86
		2	87
		3	88
		4	88
		5	87

Berdasarkan hasil akurasi pada Tabel 6.5 diatas, dibuat tabel rata-rata nilai hasil akurasi dengan data latih sebanyak 20, 30 dan 40. Rata-rata akurasi dari pengujian jumlah data latih ditunjukkan pada Tabel 6.6.

Tabel 6.6 Rata-rata hasil pengujian pengaruh jumlah data latih

Data Latih	Data Uji	Rata-rata Akurasi (%)
20	120	84.8
30		86.2
40		87.2

Berdasarkan hasil uji coba yang ditunjukkan pada Tabel 6.6, dapat disimpulkan bahwa jumlah data latih yang berbeda dapat mempengaruhi hasil dari akurasi sistem. Pada hasil pengujian, terlihat akurasi terendah terletak pada saat data latih berjumlah 20 yaitu sebesar 84.8%, kemudian pada saat data latih berjumlah 30 yaitu sebesar 86.2% dan yang tertinggi terletak pada saat data latih berjumlah 40 yaitu sebesar 87.2%. Berdasarkan perbedaan hasil akurasi tersebut, dapat disimpulkan bahwa semakin banyak data latih yang digunakan dalam pengujian, maka akan semakin tinggi hasil dari nilai akurasinya. Hal ini dikarenakan banyaknya data latih berdampak pada semakin banyak pula data yang digunakan sebagai pertimbangan dalam proses klasifikasi, disamping sebaran data seimbang dapat menjadi salah satu factor tingginya akurasi yang dihasilkan dalam proses klasifikasi. Berikut merupakan grafik hasil pengujian pengaruh jumlah data latih pada Gambar 6.4.



Gambar 6.4 Grafik hasil pengujian pengaruh jumlah data latih

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan dari hasil pengujian yang telah dilakukan pada penelitian ini mengenai Penentuan Penerimaan Beasiswa mahasiswa menggunakan Metode *Modified K-Nearest Neighbor* dapat disimpulkan sebagai berikut:

1. Metode *Modified K-Nearest Neighbor* dapat diimplementasikan dalam klasifikasi penentuan penerimaan beasiswa mahasiswa dengan menggunakan 5 parameter dan 2 jenis status penerimaan.
2. Berikut merupakan hasil akurasi yang didapatkan berdasarkan pengujian yang telah dilakukan.
 - A. Berdasarkan dari hasil pengujian pengaruh nilai k , nilai akurasi tertinggi sebesar 87% dan nilai k terbaik adalah $k=5$. Disimpulkan bahwa nilai k yang digunakan memiliki pengaruh terhadap hasil dari akurasi sistem. *Error* terjadi pada semakin tinggi nilai k dari nilai k terbaik menyebabkan menurunnya hasil nilai akurasi, hal tersebut dikarenakan semakin besar nilai k maka semakin banyak pula tetangga terdekat yang digunakan sebagai pertimbangan pengambilan keputusan dalam proses klasifikasi sehingga dapat menyebabkan *noise*. Dan juga, apabila semakin rendah nilai k dari nilai k terbaik maka semakin akan sedikit tetangga terdekat yang digunakan dalam proses klasifikasi sehingga berdampak pada nilai akurasi yang dihasilkan lebih rendah.
 - B. Berdasarkan dari hasil pengujian kelas seimbang dan tidak seimbang nilai akurasi tertinggi sebesar 86.6%. Didapatkan hasil bahwa data latih seimbang mengalami peningkatan akurasi yang stabil. Sedangkan pada data latih tidak seimbang hasil akurasi yang didapat cenderung tidak stabil mengalami peningkatan dan penurunan hasil akurasi. Hal tersebut disebabkan oleh pada data latih tidak seimbang terdapat satu kelas tertentu yang mendominasi pada proses klasifikasi sehingga hasil yang didapat tidak akurat.
 - C. Berdasarkan dari hasil pengujian pengaruh jumlah data latih nilai akurasi tertinggi sebesar 87.2%. Didapatkan hasil bahwa hasil akurasi meningkat seiring dengan bertambahnya jumlah data latih yang digunakan hal itu disebabkan karena semakin banyak jumlah data latih maka data yang dibandingkan dalam proses klasifikasi juga menjadi semakin banyak, hal tersebut dapat berpengaruh pada nilai akurasi yang dihasilkan.

7.2 Saran

Untuk penelitian selanjutnya, saran yang dapat diberikan berdasarkan kesimpulan dari penelitian mengenai Penentuan Penerimaan Beasiswa menggunakan Metode *Mofied K-Nearest Neighbor* ini adalah sebagai berikut:

1. Menambah parameter dalam penentuan penerimaan beasiswa seperti nilai keaktifan kegiatan mahasiswa agar variasi data yang digunakan pada proses klasifikasi menjadi lebih beragam dan dapat memberikan hasil yang lebih optimal.
2. Diharapkan adanya kombinasi metode *Mofied K-Nearest Neighbor* dengan metode lain agar nilai akurasi yang dihasilkan dapat menjadi lebih optimal dan mengurangi *error* yang terjadi.



DAFTAR PUSTAKA

- Ardika, I. W. D., 2013. Fenomena Pokok Pendidikan Indonesia: Apa dan Bagaimana. Bali: Politeknik Negeri Bali.
- Badan Pusat Statistik, 2016. 35730.1607. Statistik Daerah Kota Malang. Badan Pusat Statistik Kota Malang.
- Putri, P. M. B., 2017, Diagnosa Penyakit Kulit Pada Kucing menggunakan Metode Modified K-Nearest Neighbor. Malang: Universitas Brawijaya.
- Oswal. A., Shetty, V., Badshah, M., Pitre, R. & Vashi, M., 2014. *A Survey on Disease Diagnosis Algorithms*. International Journal of Advanced Research in Computer Engineering & Technology Volume 3 Issue 11, November 2014.
- V, V., V. & Ravikumar, A., 2014. Study of Data Mining Algorithms for Prediction and Diagnosis of Diabetes Mellitus. International Journal of Computer Applications. Trivandrum: India.
- Lestari, D., 2017. Perbandingan Klasifikasi Beasiswa Toyota Astra Menggunakan *K-Nearest Neighbor Classifier* dan *Naïve Bayes* Sebagai Penentu Metode Klasifikasi Pada Sistem Pendukung Keputusan Penerimaan Beasiswa Toyota Astra. Surabaya: Institut Teknologi Sepuluh Nopember.
- Mazda, G., 2018. *Menawan Janjikan 1.000 Beasiswa*. [online] Tersedia di: <<https://www.radarmalang.id/menawan-janjikan-1-000-beasiswa-kuliah/>> [Diakses 18 Februari 2018]
- Parvin, H., Alizadeh, H. & Bidgoli, B., 2008. *MKNN: Modified K-Nearest Neighbor*. Proceedings of the World Congress on Engineering and Computer Science 2008. San Fransisco: United Nations.
- Simanjuntak, T. H., 2017. Implementasi *Modified K-Nearest Neighbor* dengan Otomasi Nilai K Pada Pengklasifikasian Penyakit Tanaman Kedelai. Malang: Universitas Brawijaya.
- Sudarasana, I. K., 2016, Pemikiran Tokoh Pendidikan Dalam Buku Lifelong Learning: Policies, Practices and Programs. Denpasar: Institut Hindu Dharma Negeri Denpasar.
- Sulistyo, D., 2015. Pemanfaatan Informasi Teknologi Dalam Penentuan Beasiswa Siswa Kurang Mampu. Yogyakarta: Universitas Ahmad Dahlan.
- Septianto, R. H., 2015, Diagnosa Penyakit Tanaman Kopi Arabika dengan metode *Modified K-Nearest Neighbor* (MKNN). Malang: Universitas Brawijaya.
- Undang-undang Peraturan Pemerintah nomor 47 tahun 2008 tentang Wajib Belajar. Jakarta: Kementrian Sekretariat Negara Republik Indonesia
- Wafiyah, F., 2017. Implementasi Algoritma *Modified K-Nearest Neighbor* (MKNN) untuk Klasifikasi Penyakit Demam. Malang: Universitas Brawijaya.

Jain, Y. K. & Bhandare, S. K., 2011. *Min Max Normalization Based Data Perturbation Method for Privacy Protection*. International Journal of Computer & communication Technology 2011. Vidisha: Samrat Ashok Technological Institute.

Hechenbicler, K. & Schliep K., 2004. *Weighted k-Nearest Neighbor Techniques and Ordinal Classification*. Munich: ludwig maximilian university.



LAMPIRAN

A. Data Mahasiswa

No	NRP	Nama	IPK	Semester	Gaji Orang Tua	Tanggungan	UKT	Status
1	1112100014	NUR MUHAJIMIN	3.27	7	2287900	2	4000000	diterima
2	1112100067	WAHYU INDAYANI	2.93	7	4469900	5	7500000	ditolak
3	1113100116	CLARISSA A SITORUS	3.2	5	4118300	3	7500000	ditolak
4	1114100081	AHMAD RIZKI KURNIAWAN	3.25	3	2000001	2	1000000	ditolak
5	1114100107	LEVINA ALMIRA	3.14	3	7500001	2	7500000	ditolak
6	1212100070	KURNIA PUTRI PURNANINGARUM	2.88	7	2000000	2	1000000	ditolak
7	1213100048	BELLA DINDA FAMELA	2.96	5	2400000	1	2500000	diterima
8	1214100029	NUKE EVA NOVITA	3.04	3	4600000	0	4000000	ditolak
9	1214100095	DANTI ARDIANTI SOESANTO	3.5	3	30500000	2	7500000	ditolak
10	1312100010	NUR AFIFAH AMALIA	3.48	7	4445500	3	7500000	diterima
11	1312100089	FITRIA DWI WULANDARI	3.5	7	2000000	1	4000000	diterima
12	1313100001	MAUDI PRAMEDIA PUTRI	3.63	5	3000000	4	2500000	ditolak
13	1313100004	RATIH ARDIATI NINGRUM	3.46	5	4500000	1	4000000	ditolak
14	1313100010	WAHYU INDRI ASTUTI	3.35	5	4500000	2	2500000	ditolak
15	1313100044	HANIF YONTAR RAHMA	3.15	5	4601200	3	2500000	ditolak
16	1313100055	MUHAMMAD SYAUQI KHUDZAIFI	3.26	5	4599200	2	2500000	ditolak
17	1313100083	MUHAMMAD AZHAR	3.61	5	4166000	2	500000	ditolak
18	1313100121	NURSYABANI HENDRO	2.97	5	12000000	2	7500000	ditolak
19	1313100131	VINONDANG M.G.A. SINAGA	2.94	5	6150000	2	7500000	ditolak
20	1314030005	FEBRI ANITA SARI MP	3.47	3	2000000	4	2500000	ditolak
21	1314100113	DINI FITRI HARIANI	3.49	3	5469200	2	7500000	ditolak
22	1411100004	MAWADDATUL WAROCHMAH	2.75	9	2000000	4	7500000	ditolak
23	1412100010	SITI MAMLUATUS SAADAH	3.68	7	2755200	1	2500000	diterima
24	1412100017	ANDIKA PUTRI	2.67	7	2000000	0	2500000	ditolak
25	1412100018	ZILFIRDAUSI AHLA ZAKARIA	3.26	7	5000000	2	7500000	ditolak
26	1412100102	DENIDA MEGA A	2.62	7	1690905	1	2500000	ditolak
27	1413100039	VENA ZULFIAH	3.13	5	4500000	1	2500000	ditolak
28	1413100075	RANDY YUSUF KURNIAWAN	3.75	5	5000000	3	2500000	diterima
29	1512100055	INDIRA RIZQITA IVANESTHI	3.47	7	7500000	2	2500000	diterima
30	1513100013	CHUSNUL EKA SAFITRI HIMAYANI	2.87	5	1839000	1	2500000	ditolak
31	1513100064	AFRA BASYIRAH	3.23	5	5000000	4	2500000	ditolak
32	1513100074	ANNISA KHAIRA	3.02	5	4614400	1	4000000	ditolak
33	2112100159	BELLA KARTIKA FERANI	3.02	7	4347900	1	4000000	ditolak
34	2113100149	AHMAD TARMIZI	3.16	5	5495600	1	4000000	ditolak

35	2114030044	ISMAH FAWAIZ	3.21	3	2000000	1	2500000	ditolak
36	2114105033	NURHADI	2.65	3	950000	2	7500000	ditolak
37	2212100023	MUHAMMAD FATHU SUNU MUKTI	3.03	7	5057350	3	4000000	ditolak
38	2212100168	DHITYO YULIANSYAH	3.03	7	4399300	1	4000000	ditolak
39	2213100058	WILLYBRORDUS	3.49	5	4000000	1	5000000	ditolak
40	2213100084	IQDAMIL JIHAD	2.93	5	20000000	3	4000000	ditolak
41	2213100104	FEBRY ANGGA GUNAWAN	3.44	5	1500000	0	500000	diterima
42	2213100127	GALIH BUDI VIRGIANSYAH	3.26	5	1500000	1	2500000	ditolak
43	2213100341	ANAS MAULIDI UTAMA	3.83	5	5931473	3	7500000	diterima
44	2214038013	KRISNA DIAN SAPUTRA	3.19	3	1500000	4	7500000	ditolak
45	2313100020	NAUFAL NURRACHMADI RISTIANSYAH	3.37	5	4500000	3	2500000	ditolak
46	2313100028	ILHAM DITO PRASETYAWAN	3.3	5	14167900	1	7500000	ditolak
47	2313100055	ACHMAD ZULFIKAR FAWZI	3.54	5	6685000	2	4000000	ditolak
48	2313100057	BENEDICTUS IVAN MARTIN	3.77	5	10000000	2	6000000	ditolak
49	2313100087	DEFINA PARASANDI	3.51	5	2000000	0	2500000	diterima
50	2313100107	WINNY MARGARETA	3.98	5	5000000	1	5000000	ditolak
51	2314030002	PRIMA DANA WISESA	3.13	3	7897800	2	4000000	ditolak
52	2314105030	YUKH IHSANA ISTIGHFARINA FADLLA	3.37	3	5289600	2	6000000	ditolak
53	2314106030	DENNIS FARINA NURY	3.08	3	3000000	0	6000000	ditolak
54	2412100034	VINCENSUS CAHYA DWINANDA	3.34	7	5167100	3	7500000	ditolak
55	2413100014	ARDHIANSYAH WIDHI HARSONO	3.55	5	5500000	2	4000000	diterima
56	2413100040	FEBRYN PRADANA RIFANDA PUTRA	3.19	5	3500000	2	2500000	ditolak
57	2413100139	INTAN LURUH LARASATI	3.23	5	5667647	2	7500000	ditolak
58	2414100069	NIKEN ARINA PRATIWI	3.71	3	5601800	3	2500000	ditolak
59	2414106017	IKA SANTIATMA	3.25	3	4409000	3	6000000	ditolak
60	2512100046	DINI APRILIANI	3.56	7	2000000	1	2500000	diterima
61	2513100112	ARJUNA L A SIPAYUNG	3.06	5	3609935	4	2500000	ditolak
62	2513100180	CLARA BEATRIX HUTAPEA	3.26	5	15000000	3	7500000	ditolak
63	2514100092	RIZKY GIAN PRATAMA	3.58	3	2615744	3	2500000	ditolak
64	2712100029	RIZQI ILMAL YAQIN	3.74	7	8000000	4	7500000	diterima
65	2713100025	JONATHAN DIAN	3.61	5	4856100	2	4000000	ditolak
66	2713100043	MUHAMMAD SUCHAIMI	3.51	5	2500000	0	2500000	ditolak
67	2713100067	ANGGIAT RAMOS JUNIARTO	3.16	5	4380800	3	5000000	ditolak
68	2713100074	ORLANDO BANJARNAHOR	2.95	5	4441137	1	1000000	ditolak
69	2713100075	RIFKI RACHMAN KHOLID	3.41	3	2500000	3	2500000	ditolak
70	2713100103	M FIQHI DZIKRIANSYAH	3.27	7	4652600	2	2500000	ditolak
71	2914100020	VIRBYANSAH ACHMADAN N.	3.46	3	5167100	2	2500000	ditolak
72	3112100124	HANDI DESTIANNO ADHIKA	3	7	6000000	1	7500000	ditolak

73	3113041089	HERALDY BHASKARAWAN PUTRA	3.48	5	4500000	4	1000000	diterima
74	3113100002	AJI DICKY PERMANA	3.52	5	2000000	2	2500000	ditolak
75	3113100041	MOHAMAD FIKRI AULIA	3.31	5	5000000	3	2500000	ditolak
76	3113100081	KHAERUL BAHRI	3.3	3	2500000	3	6000000	ditolak
77	3114041065	MUHAMAD RIYAN SWANDANA	3.5	3	1500000	2	500000	ditolak
78	3114041068	NOVIN AGUSTINA NUR PUTRI	3.47	3	4702700	2	2500000	ditolak
79	3114100075	YAUW LEI SUJADI H	3.76	3	5000000	5	5000000	ditolak
80	3114106018	ACHMAD FAIQ ADHI ATMA	3.58	3	4395500	2	6000000	diterima
81	3212100063	HUSNA ABHARINA MULYANI	3.25	7	8702300	2	7500000	ditolak
82	3213100021	MUHAMMAD WHIBI RHOUZAN FIKRI	3.34	5	7000000	3	4000000	ditolak
83	3312100027	AULIA RODLIA FITRIANA	3.52	7	4841000	5	2500000	diterima
84	3312100069	GIAN CHRYSTIADINI	3.13	7	3000000	3	6000000	ditolak
85	3313100014	BINTANG RESPATI DWI HARNANI	3.29	5	273500	2	2500000	ditolak
86	3313100042	AFFAN MAULANA ASSIDIQY	2.85	5	20000000	4	7500000	ditolak
87	3313100111	AFIFAH YUSRINA	3.61	5	6371500	1	7500000	ditolak
88	3413100012	MUHAMMAD ANNAS WAFA ALBAQI	2.93	5	475000	2	2500000	ditolak
89	3413100070	MOCHAMMAD ZAINUL	2.82	5	500000	2	1000000	ditolak
90	3414100024	DESI PUTRI ISLAMY	2.63	3	2000000	1	2500000	ditolak
91	3414100090	LIDYA RIANI RAMLI	3.47	3	1500000	5	1000000	ditolak
92	3512100088	BAGAS TRI WIDODO	3.02	7	5000000	1	7500000	ditolak
93	3514100010	KRISTINA PUTRI	3.17	3	1000000	1	1000000	ditolak
94	3514100020	KARISMA RIZKIKA	3.15	3	5093000	3	2500000	ditolak
95	3514100047	ZAHRATU FIRDAUS	3.08	3	2000000	3	2500000	ditolak
96	3514100060	RISMA INDRIYA LUDFI	2.99	3	3500000	2	2500000	ditolak
97	3614100059	MAGHFIRAH BUNGAS MUWIFANINDHITA	3.26	3	7868100	2	4000000	ditolak
98	3713100015	IMAM GAZALI	2.85	5	750000	2	2500000	ditolak
99	3813100005	I WAYAN ADI DARMAJAYA	3.59	5	2750000	1	1000000	ditolak
100	3813100055	MASYITHAH TRISTY KINASIH	3.56	5	10716700	3	7500000	ditolak
101	4112100011	MUHAMMAD ANSORI HASIBUAN	3.15	7	8000000	5	7500000	ditolak
102	4112100055	TRY ARISMUNANDAR	3.19	7	7000000	2	5000000	ditolak
103	4112100100	TRI SUKRISNA WISNAWA	3.37	7	4000000	3	7500000	ditolak
104	4112100106	PANDU AUDITYA PRATAMA	2.87	7	7000000	2	6000000	ditolak
105	4112100115	HARISUDDIN HAWALI	3.18	7	5000000	2	7500000	ditolak
106	4113100007	MUHAMMAD WILDAN FIRDAUS	3.08	5	7000000	4	5000000	ditolak
107	4113100055	DINA AYU PUSPITASARI	2.72	5	3500000	5	500000	diterima
108	4113100073	LAURA NATHASYA	3.25	5	5000000	1	7500000	ditolak
109	4114100065	ILHAM SALO	3.54	3	5675000	8	4000000	ditolak

110	4114100067	MUHAMMAD ALIF BUDIMAN	3.59	3	2500000	3	4000000	ditolak
111	4212100023	RAHMAT KRISTOMI	3.71	7	3718700	3	6000000	ditolak
112	4212100119	GINANJAR BASUKI	3.2	7	3000000	2	2500000	diterima
113	4213100080	YUDHA ADI PRATAMA	3.59	5	7000000	2	4000000	ditolak
114	4214100037	IRSAT SURYA SEKTI	3.18	3	3192300	1	4000000	ditolak
115	4214100044	DAVID MAHENDRA PUTRA	3.3	3	4000000	2	2500000	ditolak
116	4214106006	WIWIN ROHMAWATI	3.68	3	3767800	1	6000000	ditolak
117	4312100051	WILDA NINGRUM ROFIKA RIZKI	2.95	7	1000000	1	2500000	ditolak
118	4312100070	BENEDICTUS JOHANES BELALAWA	3.4	7	4584900	2	4000000	ditolak
119	4312100073	KRESNA HADI DWIPAYANA	3.36	7	2500000	4	7500000	ditolak
120	4313100004	IRAWATI DIAH RATNA PUTRA IDA AYU	3.01	5	2500000	2	4000000	ditolak
121	4313100017	ASTRID FITRIA FEBRYANTI PRAMONO	3.5	5	3809200	1	4000000	ditolak
122	4313100028	YUDA ARIFIANTO	3.55	5	2000000	1	2500000	diterima
123	4313100043	MEI LINDAWATY	3.24	5	200000	3	500000	ditolak
124	4313100094	AULIA PUSPITORINI	3.19	5	2800000	2	500000	diterima
125	4313100097	DIAN PUSPITA SARI	3.51	5	2500000	1	2500000	ditolak
126	4313100114	GALIH RAMADHAN	3.33	5	2474000	1	500000	ditolak
127	4313100129	FAUZAN AWAL RAMADHAN	3.31	5	3286800	5	7500000	ditolak
128	4313100141	DANNY INDRA SETYAWAN	3.13	5	10764700	2	7500000	ditolak
129	4313100149	JAMHARI HIDAYAT BIN MUSTOFA	3.3	5	6699714	3	7500000	ditolak
130	4412100028	I MADE INDRA GUNAWAN	2.8	7	2550000	1	7500000	diterima
131	4412100048	DANAN LESMANA WIWASWAN	2.57	7	1500000	1	2500000	ditolak
132	4413100011	BESTI PRATIWI	3.2	5	10200000	1	5000000	ditolak
133	4414100002	SYAUGHI ALIF FADHILA	3.14	3	3143000	4	2500000	ditolak
134	4414100037	FAJAR WAHYU SAPUTRA	3.32	3	3714400	2	2500000	ditolak
135	5112100185	DWI AL AJI SUSENO	3.39	7	3500000	2	7500000	ditolak
136	5112100191	IZDIHAR FARAHDINA	3.33	7	18328200	2	6000000	ditolak
137	5113100004	IMAGINE CLARA ARABELLA	2.95	5	3400000	3	2500000	ditolak
138	5113100011	ALIYA RAHMA NAJIHATI	3.13	5	6000000	2	4000000	ditolak
139	5113100012	ALIYA FATHMA NAJIHATI	3.31	5	6000000	2	4000000	ditolak
140	5113100022	M. SYAIFUL JIHAD A	3.26	5	1220000	1	2500000	ditolak
141	5113100031	ADIAN LATIFA NURROHMAN	3.35	5	5513700	2	4000000	ditolak
142	5113100033	DAVID VICTOR GIANDLY	2.73	5	2500000	2	4000000	ditolak
143	5113100046	BURHANUDIN RASYID	3.83	5	4000000	3	2500000	ditolak
144	5113100077	ILHAM GURAT ADILLION	3.62	5	6000000	1	5000000	ditolak
145	5113100078	RICHARD ALVIN SIANTURI	3.13	5	3570225	2	4000000	ditolak
146	5113100100	NIDA AMALIA	3.65	5	8000000	5	5000000	ditolak
147	5113100148	JULIO A LEONARD	3.77	5	7871000	3	4000000	diterima

148	5113100153	SANI PUJI RAHAYU	3.6	5	5748037	2	5000000	ditolak
149	5113100166	IHSAN PRASETYA	3.25	5	7373203	2	7500000	ditolak
150	5114100092	KHARISMA MONIKA DIAN PERTIWI	3.53	3	5000000	3	5000000	ditolak
151	5114100114	SHAFLY NAUFAL ADIANT	3.21	3	6000000	2	4000000	ditolak
152	5213100108	KEVIN SETIAWAN	3.54	5	8000000	2	5000000	ditolak
153	5213100134	M ASVIN IMADUDDIN	3.48	5	4500000	1	4000000	ditolak
154	5213100190	DENNY FASIHUL IKSAN	3.47	5	3000000	3	7500000	ditolak
155	1113100096	PAMBAYUN PURBANDINI	3.42	7	6842500	1	4000000	diterima
156	1113100112	SHELLY PERMATA SARI	3.11	7	3000000	1	7500000	ditolak
157	1114100007	TRI ILMA SARI	3.59	5	2500000	3	2500000	ditolak
158	1114100023	ADDINAL FIKRIYA	3.08	5	3300000	1	2500000	ditolak
159	1114100081	AHMAD RIZKI KURNIAWAN	3.26	5	2500000	1	1000000	ditolak
160	1114100085	OKINAWA RIKENATA AIDA ISLAMIYAH	3.01	5	3518500	1	5000000	ditolak
161	1114100094	ANNISA NURUL AINI	3.15	5	4850000	7	4000000	ditolak
162	1213100002	CHUSNUL CHOTIMAH	3.26	7	1757000	1	500000	ditolak
163	1213100004	WIDYA ARRYA SEPTIANA	3.11	7	3636500	1	2500000	ditolak
164	1213100007	DESNA YUANDA	3.05	7	3039900	1	4000000	ditolak
165	1213100010	AZARIA ELVINAROSA	3.41	7	10000000	3	2500000	ditolak
166	1213100019	SEPTIA MARGA DARTIKA	3.45	7	2795000	1	1000000	ditolak
167	1213100031	LAILATUR ROSYIDAH	3.19	7	3182000	2	1000000	ditolak
168	1213100046	CHYNTIA KUMALASARI PUTERI	3.48	7	1300000	1	5000000	ditolak
169	1213100080	PRIMA ADITYA	3.58	7	4000000	2	5000000	ditolak
170	1213100081	DINDA ULIMA RIZKY YANI	3.44	7	5425690	2	2500000	ditolak
171	1213100089	BRIGITA SANCE	3	7	4000000	3	4000000	ditolak
172	1213100111	DINAN FAKHRANA RAMADHANI	3.04	7	10219547	2	7500000	ditolak
173	1214100019	DURROTUN NUZULA FI SHOUMI	3.37	5	5000000	2	4000000	ditolak
174	1214100032	DINAH RAZAN ANSHORI	2.86	5	2060000	3	1000000	ditolak
175	1214100033	I GUSTI AYU RISKA SRI ASTARI	3.81	5	5000000	2	2500000	diterima
176	1313100014	JULIYANTO	3.33	7	1000000	2	4000000	ditolak
177	1313100022	AGENG PRAMESTHI	3.45	7	3246500	2	2500000	ditolak
178	1313100044	HANIF YONTAR RAHMA	3.34	7	4601200	3	2500000	ditolak
179	1313100067	SENDY ERLINDA SAPUTRI	3.6	7	2000000	2	500000	ditolak
180	1313100069	DEVI LINDASARI	3.46	7	2000000	1	500000	ditolak
181	1313100072	SISKA IBNI HABIBAH	3.37	7	5969554	1	4000000	ditolak
182	1313100089	SEPTO NAINGGOLAN	3.61	7	3000000	5	2500000	ditolak
183	1313100127	SYARAH PUTRI YUTIKA	3.46	7	1090000	2	7500000	ditolak
184	1314030014	ALBERTUS EKA PUTRA HARYANTO	3.8	5	2600000	2	2500000	ditolak
185	1314100014	AFI ROCHMATURIZA	3.77	5	2500000	1	2500000	ditolak
186	1314100059	SITI AISYAH	3.16	5	4000000	4	4000000	ditolak

187	1314100077	EKA AULLYA RISMA HAQQI	3.77	5	7013300	0	4000000	diterima
188	1314100092	ERLIN SUKMAPUTRI	3.36	5	3000000	2	1000000	ditolak
189	1314100095	ZUHROFATUL ULWIYAH	3.71	5	1700000	2	1000000	ditolak
190	1314100110	NIKITA DWIE SEPTIANA	3.36	5	7000000	3	7500000	ditolak
191	1314100113	DINI FITRI HARIANI	3.63	5	5626000	2	7500000	ditolak
192	1413100002	WINNA NARASIH	3.23	7	4805852	1	4000000	ditolak
193	1413100015	RIFKA ETRIANA	3.1	7	5224000	1	2500000	ditolak
194	1413100020	NUR AZIZAH AGUSTINA	3.26	7	4000001	3	2500000	ditolak
195	1413100032	TAUFIK QODAR ROMADIANSYAH	3.4	7	8509666	2	5000000	ditolak
196	1413100040	ALDILA MEGA TRIANITA	3.39	7	4000000	0	5000000	ditolak
197	1413100066	MUHAMMAD FAQRI FAHROZI HUZEIN	3.38	7	4000000	3	4000000	ditolak
198	1413100073	AHMAD UBAIDILLAH IHSANY	3.44	7	5103900	5	4000000	ditolak
199	1413100075	RANDY YUSUF KURNIAWAN	3.77	7	1000000	2	2500000	diterima
200	1413100101	LUTHFI KHOIRUN NAASHIHAAH	2.99	7	4522500	1	7500000	ditolak
201	1413100113	RISTIA RIZKY AMALIA WAHIDAH	3.31	7	7000000	1	7500000	ditolak
202	1413100114	EVRIZAL PRATAMA	3.56	7	3996447	3	2500000	ditolak
203	1414100041	TISA WAHYU HIDAYATI	3.61	5	2065350	2	2500000	ditolak
204	1414100048	NABILA FAUZIAH FARDANI	3.45	5	3000000	1	2500000	ditolak
205	1414100053	ARAS SUCI WIJAYANTI	3.27	5	5080300	3	4000000	ditolak
206	1414100074	NANIK ISMI OKTAVIANI	3.74	5	3553000	2	1000000	ditolak
207	1414100084	YULINAR DWI NUR AZIZAH	3.53	5	1800000	2	500000	ditolak
208	1513100014	MUHAMMAD EVAN NURRAHMAWAN	3.62	7	5559948	2	4000000	ditolak
209	1513100064	AFRA BASYIRAH	3.39	7	5000000	2	2500000	ditolak
210	1514100033	TIARA PUJI ANDAYANI	3.33	5	3037500	1	4000000	ditolak
211	1514100057	AINI IKA RAHMAWATI	3.43	5	5035600	3	4000000	ditolak
212	1514100075	IZZA NUR LAILY	3.64	5	4000000	1	5000000	ditolak
213	2113100013	NIMAS ALFIANA RARAS SAPUTRI	3.04	7	3600000	1	4000000	ditolak
214	2113100125	ANANG SAPUTRA PANJAITAN	3.16	7	2500000	1	500000	ditolak
215	2113100139	KEVIN KURNIAWAN S	3.74	7	4000001	2	4000000	diterima
216	2114030024	DIMAS WISNU WARDANA	3.07	5	7000000	2	4000000	ditolak
217	2114030067	RAMADHANA KURNIA PUTRA WIJAYANTO	3.05	5	4100000	2	7500000	ditolak
218	2114100004	ROVITA HARIM BHARADELA FITRIANA	3.18	5	6200000	1	4000000	ditolak
219	2114100016	FAUZI PERDANA	3.35	5	8700000	2	4000000	diterima
220	2114100036	BASKORO ARDY KUSUMA	3.17	5	3034000	3	4000000	ditolak
221	2114100068	NOVITA NUR WULANDARI	3.56	5	3100000	2	1000000	diterima
222	2114100079	MAULIKA GUSTINA JAISYAH	3.15	5	8601020	3	4000000	ditolak
223	2114100146	ELMAN JOKO TRIANTO	3.29	5	5700000	1	2500000	ditolak

224	2213100028	RIZKY NAFIAR R	3.47	7	4814000	1	5000000	diterima
225	2213100042	RIZKAURUM NUR FADLIYAH	3.41	7	4254100	1	1000000	diterima
226	2213100064	AKBAR DWI SYAHPUTRA	3.26	7	12748100	2	6000000	ditolak
227	2213100075	JATU MONTAZERI	3	7	3750000	2	4000000	ditolak
228	2214030078	FARIDA HERNING TYASTUTI	3.48	5	4200000	3	4000000	ditolak
229	2214100111	RIZQI PUTRA M	3.38	5	30000000	2	7500000	ditolak
230	2214100121	VERDIANSYAH	3.3	5	3520000	0	2500000	ditolak
231	2214100125	WILDAN BRAMAJI	3.18	5	17326863	2	6000000	ditolak
232	2214100133	AHMAD NAUFAL FIRDAUS	3.18	5	2554500	3	2500000	ditolak
233	2214100197	BORIES YUDO SATRIO	3.46	5	10500000	3	7500000	ditolak
234	2215105005	WISNU FAJRI	3.74	3	1885000	2	6000000	ditolak
235	2215105035	ALIN WAHYU SETIAWAN	3.6	3	2000000	2	6000000	ditolak
236	2313100024	NORA ELLYZA	3.19	7	1650000	3	4000000	ditolak
237	2313100048	RENOVA PANJAITAN	3.43	7	5000000	4	2500000	ditolak
238	2314100007	USWATUN DASIH PRAMUDYAWATI	3.64	5	4601800	2	4000000	ditolak
239	2314100107	AINURAZIS RAMADHANA	3.65	5	5000000	3	6000000	ditolak
240	2413100031	VERDANA MARCHIAN ILHAM PUTRA	3.51	7	4356600	1	2500000	diterima
241	2413100130	FEBIANTI AMIRA RAHMADANI	3.74	7	6210500	2	7500000	diterima
242	2414100068	PIETER KARUNIA DEO	3.71	5	2500000	2	500000	ditolak
243	2416105024	JERRY ARDIYANTO	3.33	1	3000000	5	7500000	ditolak
244	2513100007	RIZA NUR MADANIYAH	3.5	7	899000	2	500000	ditolak
245	2513100040	FIKI APRILIA VENA	3.52	7	5078700	1	1000000	diterima
246	2513100044	RENATA FANI JULI AGUSTI	3.45	7	2000000	1	4000000	ditolak
247	2513100061	DWIKA PUSPA WARDHANI	3.67	7	3000000	1	2500000	diterima
248	2713100001	MOHAMMAD RIZKI FEBRIANTO	3.58	7	5167100	2	2500000	ditolak
249	2713100040	FERDIANSYAH IQBAL RAFANDI	3.85	7	6000000	2	5000000	diterima
250	2713100115	ANNISA RIYANTI	3.52	7	8000000	1	4000000	ditolak
251	2713100143	AI SYAH JATI PUTRI	3.63	7	3600000	2	7500000	ditolak
252	2714100011	JODIE STEVEN	3.5	5	4500000	1	2500000	ditolak
253	2714100034	LUTFI YUNUS WAHAB AL-AUDHAH	3.41	5	3974954	3	4000000	ditolak
254	2714100137	RENDI	3.66	5	4000000	1	7500000	diterima
255	2914100008	KADEK SIERLY SARASWATI	3.37	5	1645039	2	2500000	ditolak
256	3113041020	REZA PAHLEVI ZAIN NURIFAWAN	3.24	7	13548739	2	6000000	ditolak
257	3113041029	HERTA AHSANI TAKWIM FATONI	3.34	7	5167100	1	5000000	ditolak
258	3113041071	EKKY HARDIYANTO	3.39	7	5000000	2	2500000	ditolak
259	3113100053	HANDI FIRMAN SYAH RAHMANANTA	3.55	7	4186900	2	1000000	ditolak
260	3113100074	RIFKA AMALIA	3.86	7	3672800	3	2500000	ditolak
261	3113100083	FEBRIJAYANTO DONY SAPUTRA	3.77	7	5517700	2	1000000	diterima

262	3113100094	MOHAMMAD IRWAN KURNIAWAN	3.36	7	3500000	2	1000000	ditolak
263	3113100100	ADITYA DANAR WICAKSONO	3.24	7	5850000	2	5000000	ditolak
264	3113100111	KHARISMA AGUNG	3.31	7	1500000	2	2500000	ditolak
265	3113100123	REZALVI INDRA PRANATA	3.38	7	1129000	1	7500000	ditolak
266	3113100135	MADE GITA PITALOKA	3.58	7	11800000	1	7500000	ditolak
267	3113100142	ARY WAHYUDI	3.14	7	3700000	2	7500000	ditolak
268	3114041002	M RICHZAD PRIMA S	3.42	5	8930674	3	4000000	ditolak
269	3114041003	ACHMAD WASIUR RIZQI	3.47	5	5000000	1	2500000	ditolak
270	3114041005	WAHYU SATYANING BUDHI	3.43	5	5475000	1	4000000	ditolak
271	3114041014	ANDINI DIAH PUSPANINGRUM	3.65	5	3148500	2	2500000	ditolak
272	3114041018	FRYAN CHENTYO	3.63	5	3404300	2	2500000	ditolak
273	3114041025	WINDY UNTARI SUNYOTO	3.55	5	6500000	3	5000000	ditolak
274	3114041039	OKTAVIA MERRIANDY AYU PRAHARA	3.4	5	3000000	2	5000000	ditolak
275	3114041042	FANDY ADRIAN	3.25	5	4405600	3	2500000	ditolak
276	3114041043	INDRIANI RETY HABSARI	3.62	5	7904800	1	5000000	ditolak
277	3114100008	ALKAHFAN RAMADHANI WIASANTO	3.36	5	8000000	1	5000000	ditolak
278	3114100012	MUHAMMAD ADNAN	3.4	5	8000000	2	6000000	ditolak
279	3114100014	SHARFINA CINTANTYA PURWANDANI	3.53	5	5000000	2	4000000	ditolak
280	3114100015	IQBAL ALBANA AKBAR	3.67	5	9000000	2	4000000	ditolak
281	3114100031	MUH FIKRI ARDWIAN	3.2	5	5107800	2	4000000	ditolak
282	3114100045	ZHARFAN MUHAMMAD DZULFIQAR	3.35	5	11975000	3	4000000	ditolak
283	3114100059	RIO CHANDRA	3.76	5	2000000	3	1000000	ditolak
284	3114100072	M. SHAHIB AL BARI	3.61	5	3024600	2	2500000	ditolak
285	3114100094	RIKY DWI PRASETYO	3.43	5	6000000	3	4000000	ditolak
286	3114100098	ARDELIA ARLIMASITA	3.32	5	8849200	1	5000000	ditolak
287	3114100099	MARGARETHA MUNTHE	3.65	5	5000000	4	4000000	ditolak
288	3114100106	NINDYONAWI PRADIPTO	3.59	5	3500000	2	2500000	ditolak
289	3114100145	ULY IMEGA SATRYANGGI	3.28	5	1000000	2	7500000	ditolak
290	3213100006	AHMAD RIZAL SHOLIKHUDDIN	3.36	7	4000000	1	2500000	ditolak
291	3213100008	MAULYSA PRAHASTUTI	3.46	7	4352100	2	5000000	ditolak
292	3213100012	ARMEINDA NUR AINI	3.54	7	7774100	2	4000000	ditolak
293	3213100043	KUSMIRA OKVI KARENZA	3.43	7	2500000	2	2500000	ditolak
294	3213100045	TUESDAYANI SADU	3.14	7	2500000	1	2500000	ditolak
295	3213100341	Nadya Oktavianty	3.57	7	11536685	1	7500000	ditolak
296	3214100076	WIWIT MARYADI	3.38	5	5000000	2	5000000	ditolak
297	3313100026	QORY CONSTANTYA	3.43	7	2128275	1	1000000	diterima
298	3313100045	VALENCIA SAFIR MAHARANI	3.34	7	2750000	1	1000000	ditolak
299	3313100059	ADHEE MOHAMMAD RIZKY	3.28	7	5000000	1	5000000	ditolak

300	3313100073	ICHWAN RAHMAWAN WIDODO	3.25	7	2959100	2	1000000	ditolak
301	3313100117	RACHMI LAYINA CHIMAYATI	3.25	7	5600000	2	7500000	ditolak
302	3413100055	SAKTIA GOLDA SAKINA DEWI	3.17	7	5163579	4	2500000	ditolak
303	3513100001	MUHAMMAD GHILMAN KAMAL M.	3.23	7	2500000	2	2500000	ditolak
304	3513100004	EVASARI APRILIA	3.33	7	4500000	2	4000000	ditolak
305	3513100011	CHANDRA WIDI PERMANA	3.33	7	7000000	2	5000000	ditolak
306	3513100037	ROSSITA YULI RATNANINGSIH	3.12	7	4222300	2	2500000	ditolak
307	3513100056	ANDYRA YAHYA NUGRAHA PUTRA	3.24	7	1712456	2	2500000	ditolak
308	3513100069	NURUL TAZAROH	3.05	7	5991000	2	4000000	ditolak
309	3514100022	RINO HARMASDIYONO	3.3	5	5500000	0	4000000	ditolak
310	3613100056	ASTARINA CLEOSA D	3.34	7	6329900	3	4000000	ditolak
311	3613100071	SHAFIRA AULIA ROSYIDA IRAWAN	3.66	7	16163005	2	5000000	diterima
312	3814100009	YASHINTA DIANIS LESTARI	3.46	5	2500000	0	2500000	ditolak
313	4113100028	DEDI DWI SANJAYA	3.55	7	1750000	2	5000000	ditolak
314	4113100060	ZAKI RABBANI	3.62	7	20000000	2	6000000	diterima
315	4113100067	SHAKINA AZIZUL HAQUE	3.5	7	1500000	1	500000	ditolak
316	4113100079	KURNIA NURHARDIANSAH	3.26	7	1000000	2	7500000	ditolak
317	4114100011	MOHAMMAD ADAM KHARISMA	3.48	5	3970000	2	2500000	ditolak
318	4114100034	MUHAMMAD RIFQI AUFAR	3.4	5	8887478	4	4000000	ditolak
319	4114100037	NOVARIO PRATAMA ADIGUNA	3.5	5	6136483	2	4000000	ditolak
320	4114100067	MUHAMMAD ALIF BUDIMAN	3.51	5	2500000	1	4000000	ditolak
321	4213100001	DARWIN SETIYAWAN	3.13	7	2000000	3	500000	ditolak
322	4213100002	AHMAD DWI ARTA JE MAFERA	3.5	7	4373500	1	4000000	ditolak
323	4213100018	FIRMAN AZIZ NUGROHO	3.43	7	21000000	4	7500000	ditolak
324	4213100022	DYAH NOVITA SISWO	3.33	7	8937700	2	4000000	ditolak
325	4213100044	RIZKY PRADITYA ARDIAN	3.33	7	2431886	2	4000000	ditolak
326	4213100077	MUHAMMAD BAHRU SHOLAHUDDIN	3.39	7	7648400	2	4000000	ditolak
327	4213100084	STALLINE PANGESTU	3.49	7	4800000	2	4000000	ditolak
328	4213100085	SARAH FITRIYANA	3.5	7	5244150	1	4000000	diterima
329	4213100107	ADITYA ADI PRABOWO	3.04	7	20000000	2	7500000	ditolak
330	4213101038	JUDA IMANUEL OSVALDO PANGGABEAN	3.73	7	6700000	2	7500000	diterima
331	4214100073	TEUKU RACHMATTRA ARVISA	3.23	5	3000000	3	5000000	ditolak
332	4214100086	RAYZEELADITA A W	3.39	5	2052000	2	1000000	ditolak
333	4214100111	FATIYA INDRIANA SARASVATI	3.26	5	8700000	1	7500000	ditolak
334	4313100021	FEBRIAN RAMADHANA SASTRADIMAJA	3.54	7	2000000	1	2500000	ditolak
335	4313100096	GIRANG SETYO MARINDA	3.43	7	1500000	2	2500000	ditolak

336	4313100097	DIAN PUSPITA SARI	3.38	7	2500000	1	2500000	ditolak
337	4314100008	NADYA RAHMI MAHARANI	3.49	5	2500000	2	2500000	ditolak
338	4314100028	PRIYO IMAM WICAKSONO	3.62	5	2633000	2	4000000	diterima
339	4314100040	DWI LESTARI	3.34	5	10000000	4	5000000	ditolak
340	4314100047	SOFYAN WAHYU WIDHESTOMO	3.31	5	8000000	2	4000000	ditolak
341	4314100082	HENDI DARIO W	3.08	5	8000000	1	5000000	ditolak
342	4413100010	CHANDRA KARTA YUDHA	3.34	7	4476662	2	2500000	ditolak
343	5113100002	I NYOMAN PANDE WAHYU D	3.56	7	3500000	1	5000000	ditolak
344	5113100051	MARDIANA SEKARSARI	3.7	7	9757800	0	5000000	ditolak
345	5113100073	I PUTU GEDE INDRA GUNAWAN	3.62	7	2400000	1	500000	ditolak
346	5113100102	I PUTU DWI PRATAMA ARIJAYA	3.68	7	5000000	2	5000000	ditolak
347	5113100153	SANI PUJI RAHAYU	3.58	7	5748037	2	5000000	ditolak
348	5114100002	WILLIAM SUHUD	3.24	5	2850000	2	4000000	ditolak
349	5114100010	RIVALDO ALIF	3.15	5	10600000	1	4000000	ditolak
350	5114100025	I PUTU EKA WIRA MAHARDIKA	3.4	5	7000000	2	5000000	ditolak
351	5114100043	JEFFRY NASRI FARUKI	3.67	5	3500000	1	2500000	ditolak
352	5114100044	RANI AULIA HIDAYAT	3.71	5	20000000	2	6000000	ditolak
353	5114100053	TIONIA RIZKIK AFI LA HARYONO	3.64	5	7190000	1	4000000	ditolak
354	5114100062	LUCHA KAMALA PUTRI	3.52	5	10000000	1	5000000	ditolak
355	5114100067	DZAKY ZAKIYAL FAWWAZ	3.56	5	5154500	0	2500000	ditolak
356	5114100075	PANJI RIMAWAN	3.81	5	5455000	1	5000000	ditolak
357	5114100076	MUHAMMAD FAISHAL ILHAM	3.46	5	3200000	2	4000000	ditolak
358	5114100090	NURUL AZIZAHĀ	3.21	5	7410000	6	4000000	ditolak
359	5114100116	HUMAIRA TRI ACANTYA	3.11	5	5335300	2	1000000	ditolak
360	5114100151	RIYADLATIN NUFUS	3.44	5	1500000	2	1000000	ditolak
361	5114100171	GLLEEN ALLAN M.	3.38	5	15000000	2	7500000	ditolak
362	5115100162	ASTRID FEBRIANCA	3.67	3	22736299	1	7500000	ditolak
363	5213100023	NANDA PUJI NUGROHO	3.46	7	5120900	1	4000000	ditolak
364	5213100026	MOHAMMAD RIZZA FIRMANSYAH	3.46	7	3482471	2	4000000	ditolak
365	5213100083	NURITA DAMAYANTI	3.59	7	4396000	3	2500000	ditolak
366	5213100185	SARAH PUTRI RAMADHANI	3.76	7	4000000	0	7500000	diterima
367	5214100021	STANLEY WIJAYA	3.69	5	4500000	0	4000000	ditolak
368	5214100062	NAUFAL RAIHAN NOLY	3.22	5	5287137	2	4000000	ditolak
369	5214100130	RAGESA MARIO JUNIOR	3.22	5	7900000	2	6000000	ditolak
370	5214100147	M AMMAR FAUZAN	3.69	5	5000000	2	4000000	diterima